Energy Efficiency in Green Internet of Things (IoT) Networks

Laith Kadhim Farhan

PhD 2020

Energy Efficiency in Green Internet of Things (IoT) Networks

Laith Kadhim Farhan

A thesis submitted in partial fulfilment of the requirements of
Manchester Metropolitan University for the degree of
Doctor of Philosophy

Department of Engineering
Manchester Metropolitan University

2020

Table of contents

List of figures

List of tables

Abstract

Internet of Things (IoT) is having a major impact on the digital world and how we interact with the internet. The wireless sensor network (WSN) is a promising wireless communication system for enabling IoT networks. But these networks have limited energy (battery) resources and energy-saving has become a pressing need in such networks and there have been increasing efforts to minimise energy consumption via message scheduling, optimal routing, clustering formation, aggregation techniques, etc. However, significant improvement is still required and this study has produced algorithms which have been shown to reduce energy consumption and prolong network life.

Increasing the number of neighbour nodes around a node has a negative impact on the network lifetime of WSNs. This is due to the adverse effects caused by overhearing and interference. This thesis presents a new routing technique that considers the transmission distances from one node to all neighbouring nodes within its transmission range. The interference measurement approach is adopted to select the next-hop node. The cluster head (CH) node selection is based on transmission distances to the base station (BS) with the nearest node to the BS in a sub-cluster elected as CH node for that sub-cluster. The thesis also introduces a novel scheduling algorithm called the "long hop" (LH) which assigns high priority to messages coming from sensor nodes that are located farthest away and have accessed a high number of hops, to be served first at CH nodes. This minimised energy consumption caused by the retransmission process.

Redundant data increases the unnecessary/unwanted processing and transmission of data. Thus, the thesis introduces a new method that reduces redundant data transmission and lowers the communication costs related to sending unnecessary data. The study also provides a remote monitoring system for the end-user that can check and track the performance of the sensors/IoT devices during real-time communication.

Extensive simulation tests on randomly situated WSNs show the potential of the solutions proposed in this thesis to reduce energy consumption and extend network lifetime.

Declaration

I hereby declare that no portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Laith Kadhim Farhan
2020

I would like to dedicate this thesis to my parents, brothers and sisters.

Acknowledgements

At this particular opportunity, I am grateful to Almighty ALLAH who gave me this opportunity to learn, investigate, adventure, think critically, evaluate concepts and ideas. I would express my deep gratitude to my dear parents' father and mother, and my lovely brothers, sisters, nephews, and nieces for their love, unconditional support as well as for their encouragement, without whom I would not have reached where I am. I would like to thank my country (Iraq) for giving me this opportunity to carry out this research. I also would like to thank the Ministry of Higher Education and Scientific Research, Iraqi Cultural Attaché, and the University of Diyala for supporting and sponsoring me during the study.

Thank to my supervisor, Dr. Rupak Kharel, for providing guidance throughout this study. Special thanks to my supervision team Dr. Omprakash Kaiwartya, Dr. Mohammad Hammoudeh, Dr. Bamidele Adebisi, and Dr. Umar Raza, for their full support and valuable advice during my PhD journey. Special thanks to my cousin Qusay Hameed and his wife Laura and their sons Alexander and Ali, for supporting me and I really love them as a family. Thanks to my colleagues and friends in both Manchester Metropolitan University and the University of the Diyala for the happy time and moments they have shared with me. Also, many thanks to my dear friends (Firas Ma'an, Omar Abdulwahid, Saad Saadi, Bashar Alkubaisy, Ghaith Al-Ghazal), we have had a lot of fun together and they made this step easy and absolutely memorable. Dedicated to my officemates who shared with me the good and bad times. Dedicate to my best friend for standing by my side, I am really lucky to have you in my life.

My dear friends, please forgive me if any of you is missing here.

List of publications

Journal Papers

1. Laith Farhan, Omprakash Kaiwartya, Laith Alzubaidi, Waled Gheth, Eric Dimla, and Rupak Kharel. Toward Interference Aware IoT Framework: Energy and Geo-Location-Based-Modeling. *IEEE Access*, 7:56617–56630, 2019.

2. Laith Farhan, Rupak Kharel, Omprakash Kaiwartya, Mohammad Hammoudeh, and Bamidele Adebisi. Towards green computing for Internet of Things: Energy oriented path and message scheduling approach. *Sustainable Cities and Society*, 38:195 – 204, 2018.

Conference Papers

1. Laith Farhan, Rupak Kharel, Omprakash Kaiwartya, M. Quiroz-Castellanos, A. Alissa, and M. Abdulsalam. A Concise Review on Internet of Things (IoT) Problems, Challenges and Opportunities. In *2018 11th International Symposium on Communication Systems, Networks Digital Signal Processing (CSNDSP)*, pages 1–6, July 2018.

2. Laith Farhan, Rupak Kharel, Omprakash Kaiwartya, M. Quiroz-Castellanos, U. Raza, and S. H. Teay. LQOR: Link Quality-Oriented Route Selection on Internet of Things Networks for Green Computing. In *2018 11th International Symposium on Communication Systems, Networks Digital Signal Processing (CSNDSP)*, pages 1–6, July 2018.

3. Laith Farhan, Laith Alzubaidi, Mohamed Abdulsalam, Ali J. Abboud, Mohammad Hammoudeh, and Rupak Kharel. An Efficient Data Packet Scheduling Scheme for Internet of Things Networks. In *Proc. IEEE Diyala Third Scientific Conference of Engineering Sciences and 1st Diyala International Conference of Engineering Sciences*, Jan 2018.

4. Laith Farhan, Ali E. Alissa, Sinan T. Shukur, Mohmad Alrweg, Umar Raza, and Rupak Kharel. A survey on the challenges and opportunities of the Inter-

net of Things (IoT). In *Proc. IEEE 11th International Conference on Sensing Technology*, Sydney, Nov 2017.

5. Laith Farhan, Ali E. Alissa, Sinan T. Shukur, Mohammad Hammoudeh, and Rupak Kharel. An Energy Efficient Long Hop (LH) First Scheduling Algorithm for Scalable Internet of Things (IoT) Networks. In *Proc. IEEE 11th International Conference on Sensing Technology*, Sydney, Nov 2017.

Chapters in the Books

1. Laith Farhan and Rupak Kharel. *Internet of Things Scalability: Communications and Data Management*, volume 29. Springer International Publishing, Cham, 2019.

2. Laith Farhan and Rupak Kharel. *Internet of Things: Vision, Future Directions and Opportunities*, volume 29. Springer International Publishing, Cham, 2019.

Abbreviations

Greek Symbols

λ	Lambda
μ	Mu
$M/M/1$	Queuing Model

Acronyms / Abbreviations

6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks
ADC	Analogue-to-digital
BCH	Backup Cluster Heads
BS	Base Station
CHRA	Cluster Head Recovery Algorithm
CHs	Cluster Head nodes
CPi	Custard Pi
CTS	Clear-to-Send
DEE	Distance Energy Evaluated
DSP	Dual Separate Paths
E-HEED	Enhanced-Hybrid, Energy-Efficient Distributed
EAMMH	Energy-aware Multi-hop Multi-path Hierarchy Protocol
EAMR	Energy-Aware Multi-hop Routing
EDF	Earliest Deadline First
EEDF	Energy Efficient Data Forward

EIGR	Energy-Aware Interference-Sensitive Geographic Routing
E-LEACH	Energy Balanced-LEACH
FND	First Node Death
GPIO	General-Purpose Input/Output
HDMI	High-Definition Multimedia Interface
HEED	Hybrid, Energy-Efficient and Distributed Protocol
H-HEED	Heterogeneous Hybrid Energy-Efficient Distributed Protocol
HND	Half Nodes Death
HTTP	Hyper Text Transfer Protocol
IoT	Internet of Things
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
LEACH	Low Energy Adaptive Clustering Hierarchy
LH	Long Hop scheduling algorithm
LND	Last Node Death
MQTT	Message Queuing Telemetry Transport
MS	Mobile Sink
NIC	National Intelligence Council
NJN	Nearest Job Next
OLMS	Optimal Location for a Mobile Sink
PEGASIS	Power-Efficient Gathering in Sensor Information Systems
PER	Packet Error Rate
PNDD	Predicted Non-Dispatch Data
QoS	Quality of Service
RasPi	Raspberry Pi

RFID	Radio Frequency Identification
RF	Radio Frequency
RTS	Request-to-Send
SD	Secure Digital Card
SPLL	Shortest Path Fewest Links
SPT	Short Process Time Algorithm
TBMS	Tree-Based Mobile Sink
TCP/IP	Transmission Control Protocol and the Internet Protocol
TR	Transmission Range
USB	Universal Serial Bus
WSNs	Wireless Sensor Networks

Chapter 1

Introduction

This chapter provides a brief description of the Internet of Things (IoT), and the wireless sensor networks (WSNs) technology that enabled the IoT revolution. The chapter focuses on energy conservation as one of the major challenges facing IoT and WSNs, and identifies factors that affect energy consumption in such networks. It also discusses the novelty of this thesis, and presents the research challenges addressed in the thesis. The aim and objectives are presented, as are the novel aspects of the thesis and the contents of the chapters contained in the thesis.

1.1 Motivation

INTERNET of Things (IoT) is becoming ever more pervasive in everyday life, connected an ever greater array of diverse physical objects. The key vision of the IoT is to bring a massive number of smart objects together in integrated and interconnected heterogeneous networks, making the internet even more useful [1]. It is a futuristic paradigm where all possible devices will interact with each other regardless of their size, computing power and network connectivity, in a seamless environment [2]. It makes applications smart by sensing, data harnessing, and decision-making towards actions mostly without human intervention. IoT-enabled devices are growing exponentially, including such areas as fashion and clothing, kitchen appliances, connected cars, and wearable health-care devices [3]. According to a forecast by the Cisco Systems and other study by [4], the growth in connected devices is expected to increase from "i.e., 25 billion in 2015 to 80 billion by 2025".

IoT and other enabling technologies will have a significant impact on, for example, information gathering over large geographical areas for applications such as environmental monitoring, agriculture, industry, and surveillance [5]. It is highlighted that a massive number of objects will be enabled with the realisation of an IoT ecosystem in any geographical area. In such systems, large numbers of connected devices will sense and transmit huge amounts of data, each concerning its local environment [6] and resulting in the realisation of connected-device-oriented big data.

The framework of the IoT is based on several enabling technologies including wireless sensor networks (WSNs), cloud computing, machine learning, and peer-to-peer systems [3]. WSN is the most crucial part of the communication process of the IoT networks. It consists of number of sensor nodes that are responsible for collecting key information, performing some computations, and communicating wirelessly [5]. These nodes can be deployed over a large geographical area and generally configured in a mesh network, ultimately sending a large volume of data to a base station (BS) or a gateway, and are usually forwarded via multiple hops to reach the BS [7]. These nodes are highly constrained devices with strict limitations on battery consumption and processing capabilities. Energy is considered a scarce resource for sensor nodes [8]. Sometimes it can be costly, even impossible, to exchange an energy source, e.g., networks positioned deep in the ocean, near an active battlefield or volcano [9] or simply because the large number of nodes makes it logistically impossible [10, 11].

1.2 Research Challenges

IoT will bring an enormous number of different devices and infrastructures under the same umbrella and the consequent massive growth in connectivity of 80 billion smart devices on the internet will pose a big challenge and a range of problems [4]. Each of these devices and systems will be working on their own existing protocol stacks, architectures and data formats. This will mean that the communication between these devices will not be seamless, scalable and secure.

In WSNs, sensor nodes capable of detecting the required information, performing some processing and communicating with other connected nodes are the main component of these networks. However, the life of these nodes is often restricted by being powered by a battery with a limited life, constraining processing ability, memory, and radio communications [12]. Energy efficiency is one of the most crucial issues for WSNs. Most of the energy is consumed in data processing and transmissions [13]. This means it is not rational to waste energy on protocol overheads, the transmission of unneeded data or non-optimised transmission of data packets, especially retransmissions, due to inefficient scheduling and routing algorithms. Thus, it would be prudent to design and implement systems that are designed to minimise energy consumption, and so increase the node lifetime and thus the life of the overall network.

This thesis focuses on the optimisation of the WSNs within the context of the IoT network to address a range of challenges, and specifically energy efficiency and scalability. Following are the research questions asked and which pose the research challenges addressed by this thesis:

- **Research question 1**: What constitutes the most energy-efficient solution for the IoT in the existing methods?

- **Research question 2**: How will existing network architectures and protocol stacks be able to deploy, maintain and communicate with a massive number of smart devices?

- **Research question 3**: Can scheduling algorithms perform effectively in terms of energy conservation for IoT networks?

- **Research question 4**: Can routing protocol techniques make the right decisions for sensor nodes to minimise the energy consumption of the IoT networks while forwarding data packets?

- **Research question 5**: Can a suitable algorithm be implemented within the IoT publisher device that can reduce the number of data transmissions and thus enhance data pre-processing from sender to the receiver node, minimising storage, and energy consumption?

These research questions will be answered in the course of this thesis. The first and second questions address the challenges and strategies for WSNs and IoT networks are discussed in Chapter 2. The third and fourth questions are addressed in Chapters 3 and 4, which discuss the energy hole problem in WSNs and provide an understanding of energy-aware scheduling, energy-aware routing and the factors that affect energy efficiency in WSN-assisted IoT applications. The fifth question is addressed in Chapter 5, where the experimental work consists of six sensors that gather and monitor the data from industrial setup. This method eliminates redundant data and reduces the data transmissions which, in turn, improves the overall system.

1.3 Research Aim and Objectives

1.3.1 Research Aims

This research aims to investigate and implement new routing techniques and message scheduling algorithms to minimise the number of hops and transmission distances, and thus reduce the energy consumption of IoT devices. The research also aims to develop and implement a new aggregation method that collects the data from a system (here a gas turbine engine), eliminating the duplicate data being sent to the end-user via a public cloud infrastructure. This proposed method also presents a real-time monitoring system for the end-user to assess overall system status.

The outcome of this work will reduce the number of data transmissions, minimise energy consumption and thereby extend the network lifetime. As a result, this will increase the performance of the network.

1.3.2 Research Objectives

To effectively accomplish these aims, the research objectives are:

1. To investigate various energy-efficient algorithms currently available in the field of IoT and WSNs and carry out a detailed analysis of associated problems and challenges.

2. To implement and evaluate a message scheduling algorithm that assigns a higher priority to data coming from further away, and accesses a higher number of devices to be routed first at the cluster head (CH) nodes.

3. To validate the proposed algorithms mathematically using various metrics, as well as energy consumption, to aid simulation.

4. To develop a new routing scheme that selects the best path for the data packets based on some mathematical theoretical equations. The proposed technique will

give well-balanced network traffic, shifting the traffic from overloaded nodes to other nodes with less traffic, and thereby reducing the network congestion.

5. To propose a framework of protocols that allows real-time communication between various devices (constrained and unconstrained). This method is able to reduce the number of transmitted packets and so decrease the energy consumption and thus extend the overall network lifetime.

1.4 Research Contributions

In this thesis, several novel contributions have been made and are detailed below:

1.4.1 First Novel Feature

A scheduling algorithm is an essential part of WSNs and IoT networks. Such algorithms classify the queues and assign priorities for the packets to be sent further. Therefore, scheduling messages at the CH nodes of WSNs is expected to have a significant impact on the overall energy usage of the network. In this contribution, a new scheduling technique called the Long Hop (LH) algorithm is implemented. The LH algorithm assigns a higher priority to data that has been subjected to more hops and travelled a longer distance, enabling these to be served first at the CH nodes. The contribution also introduces a new routing algorithm called shortest path and fewest links (SPFL) which forwards the data to the next-hop node with least transmission distance and least number of forwarding nodes. Thus, packets follow the least transmission distance to reach the CH nodes and the message scheduling algorithm gives preference to messages which have been subjected to more hops. This contribution is presented in Chapter 3 and reported in [14].

1.4.2 Second Novel Feature

A transmission from a source node is potentially overheard by all other neighbour nodes located within its transmission range even if these nodes are not the intended destination. Increasing the number of neighbour nodes around a node has a negative impact on the network lifetime of WSNs. This is due to the adverse effects caused by overhearing and interference.

To help overcome this problem, a new routing technique is proposed that selects the next-hop node, with less transmission distance and fewer neighbouring nodes, thus reducing interference, to forward the data to the BS. It also introduces a new clustering algorithm around a single BS that could shorten transmission distances by selecting the CH node according to its locations relative to the final destination. By doing this, the proposed scheme reduces the transmission distances from a CH node to the BS and thus helps maximise the network lifetime. The proposed scheme finds an efficient path by which to forward the data to the BS which, in turn, helps minimise the energy consumption of the entire network. This contribution is introduced in Chapter 4 and reported in [15].

1.4.3 Third Novel Feature

In this contribution, the implementation of the IoT application is investigated using sensors to gather and monitor data from a gas turbine engine and send that data to the end-user via public cloud infrastructure. The proposed study provides preliminary results on an approach for filtering the data collected by the sensors where redundant/unneeded data are tracked and removed from the transmission queue. By eliminating redundant data, it is also possible to reduce energy consumption, both in the routing and scheduling of data over the internet. This would improve the performance of the system and thereby prolong the network's working life. This new approach will allow the end-user to track and check the behaviour and condition of the sensor nodes remotely. The work is presented in Chapter 5.

1.5 Thesis Structure

This thesis is divided into six chapters. After the Introduction, the chapters are focused to answer the research questions. The thesis is organised as follows:

Chapter 2 covers the literature regarding the concepts and challenges in WSNs and IoT networks. We first introduce the characteristics and architecture of sensor networks and their applications, and then discuss reasons and solutions methods to minimise energy consumption. Future research orientations are discussed and a comparison of our study to other work is made.

Chapter 3 introduces the role of message scheduling and energy-oriented path selection algorithms by explaining their objectives and features. It shows, from the design, development and analysis of the proposed framework, that cooperation between path selection and message scheduling significantly improves energy efficiency in sensor-enabled, wireless network environments.

Chapter 4 demonstrates a new routing strategy and a distributed clustering formation. It describes the design details of the proposed protocol, including architecture, aims and protocol overview.

Chapter 5 discusses the practical application of the IoT network. The proposed system uses smart sensors, data collection/storage and cloud-based analytics connected together using the internet infrastructure to achieve the proposed system. This chapter presents the experimental tools, performance protocol, experimental setup and requirements for evaluating the solution.

Chapter 6 In this last chapter, a summary of the work carried out and the conclusions drawn from the results are presented. Suggestions for future work that would further enhance IoT networks are made.

The road map of this thesis is shown in Figure 1.1.

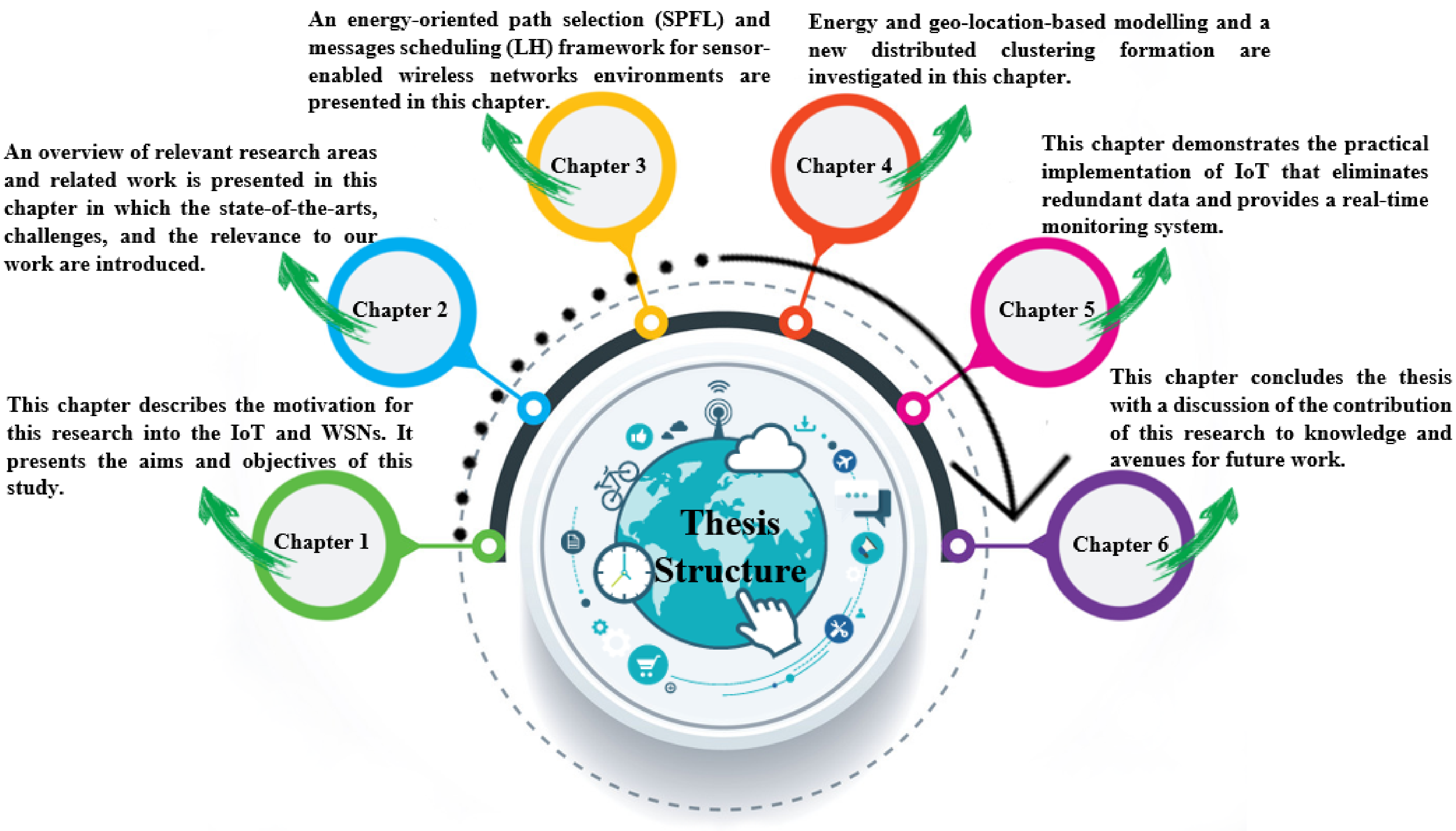


Figure 1.1 Structure of this thesis.

Chapter 2

Background and Literature Review

This chapter provides a brief introduction to the history and evolution of the internet. Then, it introduces the Internet of Things (IoT), which is followed by a list of application domains and enabling technologies. The wireless sensor network (WSN) is presented as one of the most important elements in IoT networks, and the chapter describes the relationship between WSNs and the IoT. This research is concerned with developing energy-efficiency techniques for WSNs that enable the IoT. After having identified sources of energy wastage, the chapter reviews the literature that discusses the most relevant approaches to reducing the energy consumption of IoT and WSNs. The chapter identifies a gap in the existing literature in terms of energy conservation measures that could be researched and, if successful, undertaken.

2.1 Introduction

THE internet is generally defined as a global system of interconnected computer networks that use the transmission control protocol and internet protocol (TCP/IP) to transmit and receive the data via various types of media [16]. Numerous technologies have contributed to development of the internet in its current form [17]. This has enabled more and more devices to link together and give the opportunity for these devices/things to communicate within a local network, across different networking types, and to create a much more connected world. These devices and smart objects are becoming more and more pervasive in our everyday life, and have given rise to a new concept of networking which is called the Internet of Things (IoT). Figure 2.1 shows the evolution of the internet over the last half century [17]. It started with connecting two computers together (small network) and it is moving fast to connecting billions of physical devices to the internet [18].

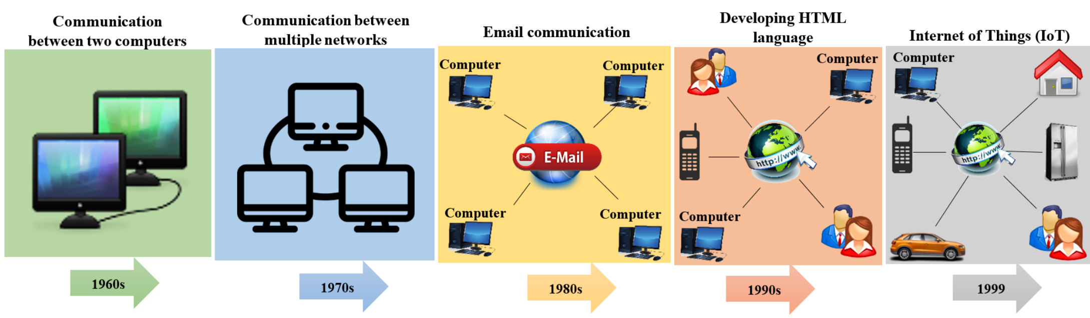


Figure 2.1 The main stages of the evolution of the internet.

The IoT is the inter-networking of physical devices used in our daily lives that utilise standard communication architectures to provide new services to the end-users [2]. The IoT brings together various emerging and enabling technologies and is changing drastically what can be achieved from the internet. The phrase "Internet of Things" was first made popular by Kevin Ashton in 1999 when he used radio frequency identification (RFID) in supply chain management [1]. Since then, the IoT has been used to define a paradigm of any and all possible devices or things that can be connected to the internet for data transfer and collection, knowledge formation and automation [2]. According to a forecast from the U.S. National Intelligence Council (NIC) in 2008 “by 2020 internet sensors may be implemented in everything such as plants, food packets, vehicles, furniture, etc.”. In the world population of 7.3 billion in 2015, there were 25 billion devices connected to the internet (i.e., 3.47 connected devices per person). This number is expected to rise to 50 billion with 6.41 connected devices per person and a world population of 7.8 billion [19]. Other study [4] reported the number of devices will reach to 80 billion in 2025 with 9.8 connected devices per person. Based on these studies, Figure 2.2 shows the growth of the IoT devices compared to the growth of world population [19–21].

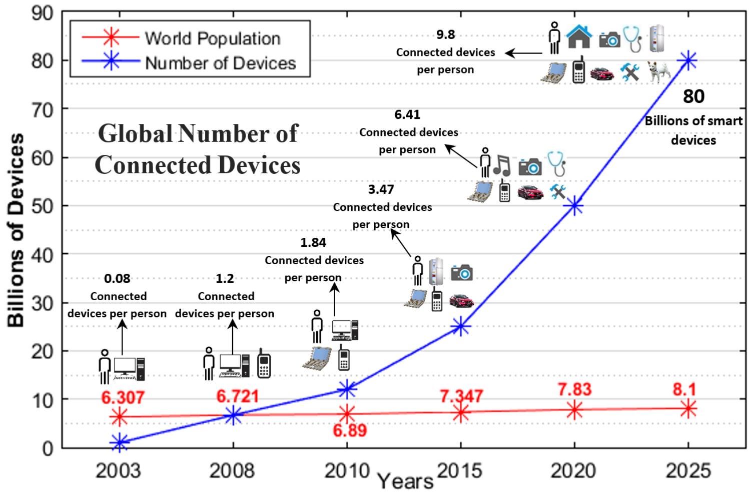


Figure 2.2 Estimated number of connected devices vs world population.

IoT networks generate a massive amount of data aggregated via these smart objects. Every two years, data doubles in size and is expected to reach 163 Zettabytes in 2025 [22]. The volume of the IoT data will grow from 2% in 2013 up to 10% in 2021 [23]. Figure 2.3 shows the growth of data from 2010 to 2025. It observes that by 2025, the volume of data will increase ten times compared to the data generated in 2016 [22, 24, 25].

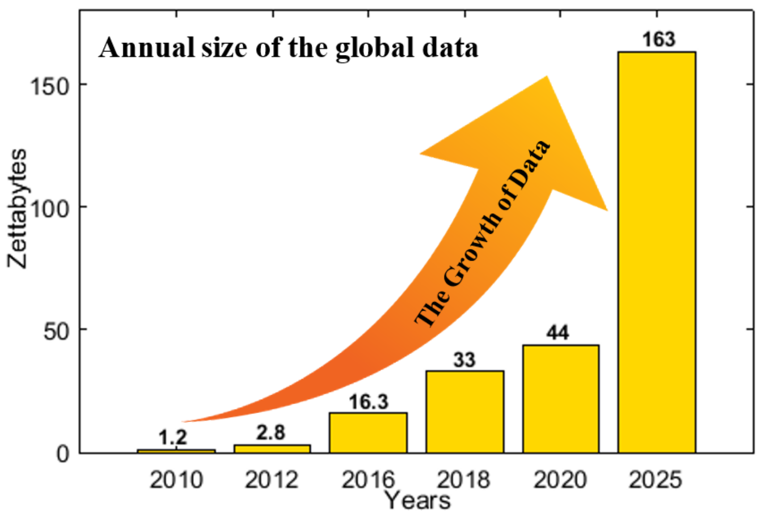


Figure 2.3 The growth of data over the years.

The concept of IoT has attracted considerable attention by governments, businesses, military, health-care, industries and researchers [3]. It can be extended to almost everything from refrigerators to washing machines, wristwatches to smartphones, home security and alarm systems, etc. [26]. For example, smart refrigerators can tell us the end of the validity of food using RFID or which items to buy during our next shopping trip to the market. Another example, users can use their smartphones or tablets with just a simple touch to control items in a house such as turning lights ON/OFF, or setting the desired temperature before getting home, this latter is now so well developed as to be advertised by APPLE, SAMSUNG, etc. as apps for their latest phones.

IoT and cloud computing both serve to increase efficiency for business and industry. The IoT generate a massive amount of data, with cloud computing providing storage and pathways for that data to travel to its destination [27]. For instance, Amazon web services are one of several IoT cloud platforms that helps people to interact with their items to be purchased through its website [28]. Also, the integration of IoT with medical technologies enables real-time monitoring system and data access to improve patient health [29]. For example, IoT devices can be used to track the real-time location of medical equipment such as wheelchairs, oxygen pumps and other monitoring equipment. Also, IoT based on WSNs technology is used to monitor a wide range of sensors in the field that are able to detect and measure various physical phenomena such as volcanic activity, flooding, and wildfires [30]. These examples are just a few applications out of thousands being developed in the field of IoT. Figure 2.4 depicts some of the areas that benefit from the IoT technology.

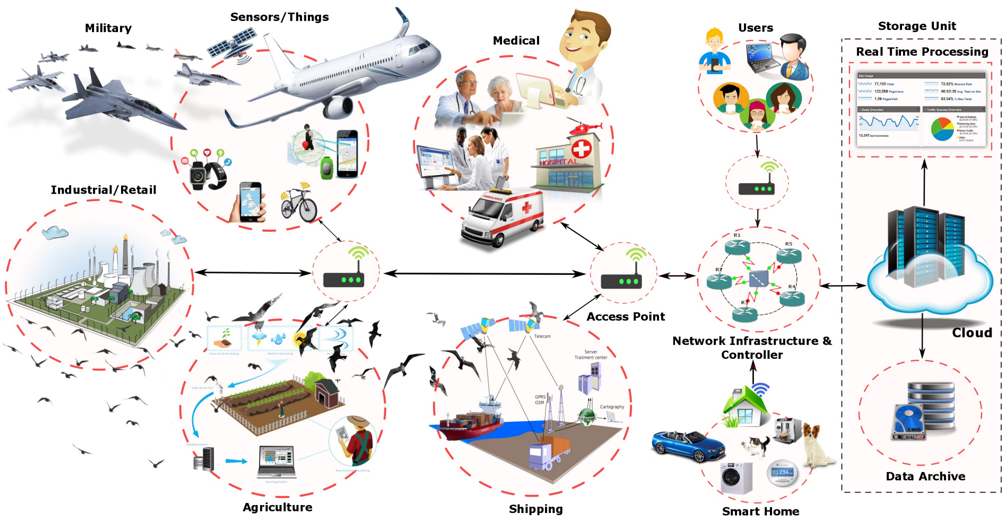


Figure 2.4 Some applications that benefit from IoT technology.

The literature review reveals that communications and IoT applications normally take the form of one of the following connections, see Figure 2.5:

- **People to People (P2P) connection:** the data transfer is from one person to the other. It occurs through video calls, telephone calls and social communications. It is usually called a collaboration connection [2].
- **Machine to People (M2P) connection:** the data transfer is from machines such as computing devices, sensor nodes or others to the users for analysis. For

example, weather forecasting uses smart devices to gather the data from the environment and send it to the administrators in the control centre for further analysis [31].

- **Machine to Machine (M2M) connection:** the data transfer is between devices without human interactions. For instance, a car talking to another car about its speed, lane change or braking intentions, etc. [32].

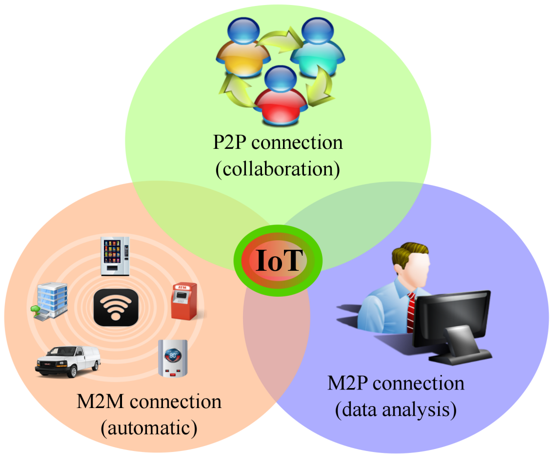


Figure 2.5 Interactions between IoT networks.

There is a large volume of published studies that cover different aspects of IoT communication. For example, the survey by Atzori, et al. [33] reported the main communication enabling technologies, both wired and wireless and the elements of WSNs. Another study [34] presented a centralised cloud vision to enable application of IoT technology to services provision. According to [2], Internet Protocol version 6 (IPv6) is the next-generation of IP, and because it will allow for more unique TCP/IP address identifiers to be generated it is an important innovation for IoT networks.

Thus, we conclude that the main components comprising IoT technology are the human element, enabling technologies and the internet. However, enabling technologies are at the heart of the IoT, which is possible only due to the development of technologies such as cloud computing, communication protocols, WSNs, embedded systems, big data analytics, mobile Internet, and web services [2].

2.2 IoT Enabling Technologies

Enabling technologies play an essential role in realising the IoT vision. These technologies provide connectivity, usability, capabilities, etc., that are required to facilitate efficient use of IoT applications [35]. There have been several studies in the literature reporting the enabling technologies, and in this section, we provide an overview of those technologies which are related to the scope of this thesis.

- **Cloud Computing:** as the numbers of connected IoT devices increase, the amount of data generated by them also increases [36]. However, IoT devices tend to suffer from limited energy, memory, processing capabilities, etc., and their integration into the cloud is the best available way to overcome most of these issues. Cloud computing is used to process, store, monitor and visualise the data comes from the IoT devices [37]. This means data processing and storage takes place in the cloud platform rather than on the IoT device [38], this has significant implications for IoT-constrained devices such as low-cost connectivity, scalability, interoperability, etc.

- **Hardware Devices:** several hardware platforms have been developed to execute IoT applications such as Raspberry Pi, NodeMCU (ESP8266), Arduino, BeagleBoard, FriendlyARM, etc. [3]. These devices vary from low-cost, low-power boards, single-boards, processing units (e.g., microprocessors, microcontrollers, etc.) and software applications that are capable of running IoT applications and communicate over the internet [39, 40].

- **Wireless Communication:** most IoT devices rely on low-power physical networking technologies such as RFID, Bluetooth, WiFi and IEEE standard 802.15.4 which are required to enable connectivity between smart devices [41]. These technologies must be globally addressable to communicate with other devices over the internet, either directly or indirectly, via an IP address [42].

- **Communication Protocol:** IoT devices require IPv4 to connect through the internet, however the near exhaustion of IPv4 addresses prior to the advent of the IoT and the prediction that there will be up to 50 billion internet connected devices by 2025 has meant that a replacement is required to permit the continued expansion of the IoT and internet in general. IPv6 is the standard proposed to replace IPv4, and uses 128-bit addressing, allowing for a total of $3.4 * 10^{38}$ unique addresses, instead of the 32-bit addressing used for IPv4 [43]. IPv6 has been applied to low-power wireless personal area networks via 6LoWPAN [44] which allows sensor nodes with limited processing ability to transmit and share their data wirelessly to the other devices/things or cloud infrastructure.

- **WSNs:** are the most crucial part of the communication process of the IoT networks. They contain sensors embedded with a microcontroller to provide intelligence and a means of communicating via the internet or some other network [45]. The sensors enable interaction with the physical world [46], and without the associated networks, there would be no bridge between the physical and virtual worlds. The benefits of connecting the WSN to the IoT is to provide remote access and allow them to communicate and exchange data with other devices and systems over the internet [5].

2.3 An Overview of WSNs

At the core of the IoT are WSNs. It is one of the most promising wireless communication systems for enabling IoT applications. WSNs consists of a various number of sensor nodes connected with each other using wireless technologies [47]. These networks are composed of low cost, low-power circuits and tiny sensor nodes. The main components of a sensor node are a power source, sensing unit, processing unit, memory and transmission unit as shown in Figure 2.6 [46].

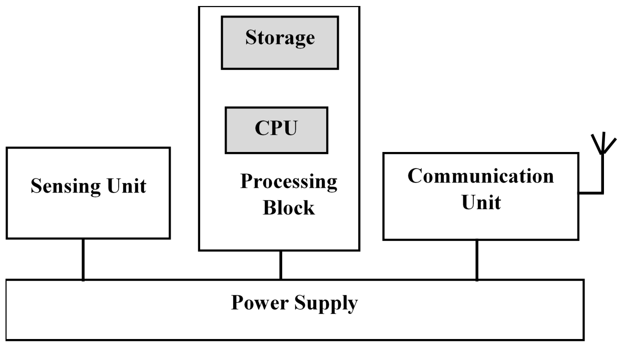


Figure 2.6 Basic elements of WSN.

These units are [46]:

- **Sensing Unit:** is the core component of the WSN and has two functions. First, it senses information from the surrounding physical environment and converts this information into digital data. Second, it forwards the data towards the processing unit.
- **Processing Unit:** contains a microprocessor with a limited amount of memory. It is responsible for receiving data from the sensing unit and sending it to the transceiver after necessary processing.
- **Communication Unit:** combines both a radio transmitter and a receiver, and is responsible for exchanging information with other sensor nodes in the sensing field.
- **Power Unit:** is responsible for supplying energy to all other units. The sensor node would die, stop obtaining and/or transmitting data if the power unit stopped working. Therefore, preserving the working life of the power unit by energy conservation becomes an important and challenging issue in WSNs.

A WSN can have many types of sensors depending on the application, whether terrestrial, underwater, underground, multimedia, or mobile [31]. The main task of the deployed sensors in WSN applications is monitoring and, as stated above can include such diverse fields as meteorology [45], fire prevention [48], flood and earthquake detection and monitoring [5], mapping environmental bio-complexity and studying environmental pollution [49]. WSNs have also been used to observe the activities of animals, birds and insects [50]. In such applications, the sensor nodes are deployed over a wide geographical region and non-accessible environments.

2.3.1 WSN Communication Architecture

A WSN is similar to a wireless ad-hoc network, since both are self-organised and multi-hop networks [51]. WSN is used to monitor and record specific phenomena and co-operatively pass data wirelessly through a gateway (base station/sink) to a central location as shown in Figure 2.7. The more modern WSNs are bi-directional (two-way communication), thus enabling control of the activity of the sensors [52].

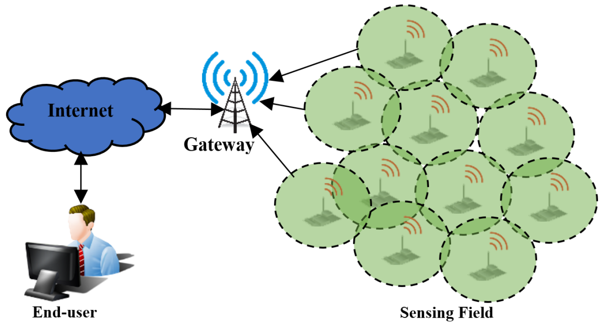


Figure 2.7 WSN communication architecture.

In large-scale networks, the sensing field is often divided into sub-clusters, with each cluster having both sensor nodes and a single cluster head (CH) node. This clustering approach has several advantages for sensor networks in terms of energy consumption, delay, network communication, etc. [53].

Sensor nodes establish connection with each other wirelessly and individually collect data from the surrounding environment, perform simple computation processes and then forward the data to its CH node via a single-hop or intermediate nodes [54]. The intermediate node serves as a data sender and path for other sensor nodes towards the CH node. These nodes make forwarding decisions (i.e., routing) based on their knowledge of the network [55]. The CH nodes can be selected randomly or based on one or more criteria such as number of neighbouring nodes, residual energy, transmission distance to the base station (BS), where the BS is the master node which collects the data from all nodes and processes it to forward to the ultimate receiver [56]. The task of the CH node is to collect data from the relevant sensor nodes, compress it and then forward it to the BS. Most of the literature on WSNs is related to a search for proper clustering, optimal path, and aggregation methods that can significantly reduce the energy consumption and lengthen the network lifetime. Figure 2.8 shows a WSN communication architecture which divides the sensing area into two clusters and each cluster has a group of sensor nodes that are connected to a single CH node and these CH nodes are connected to the BS.

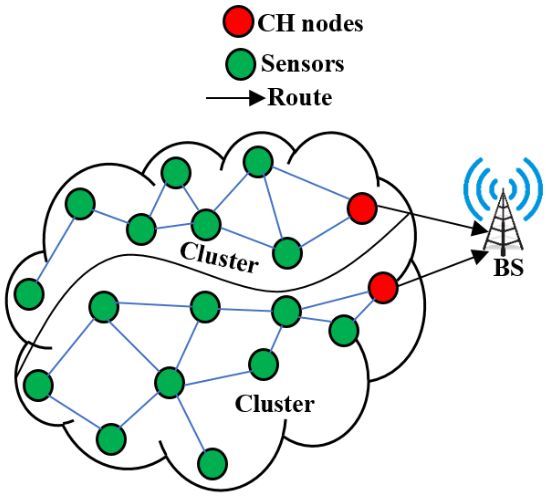


Figure 2.8 A typical clustered WSN.

2.3.2 IoT-based WSNs

The integration between WSNs and the IoT plays an important role in many applications and facilitates the universal accessibility of data, and close-to-real-time decision-making [57]. The sensor nodes connect to the internet dynamically in order to cooperate and achieve their tasks, however, most of the connected sensors are constrained within their ecosystems which have limited memories, processors and power sources [58]. When a WSN is integrated into the internet as part of the IoT, numerous decisions are required regarding that integration, including; mode of communication, hardware, computational cost, security, big data, and battery power [59, 60], see Figure 2.9. All these issues must be addressed to get the full advantages and benefits of such integration, but energy consumption is considered as one of the more important. This is due to the crucial role that these sensors play in determining the lifetime of the entire network.

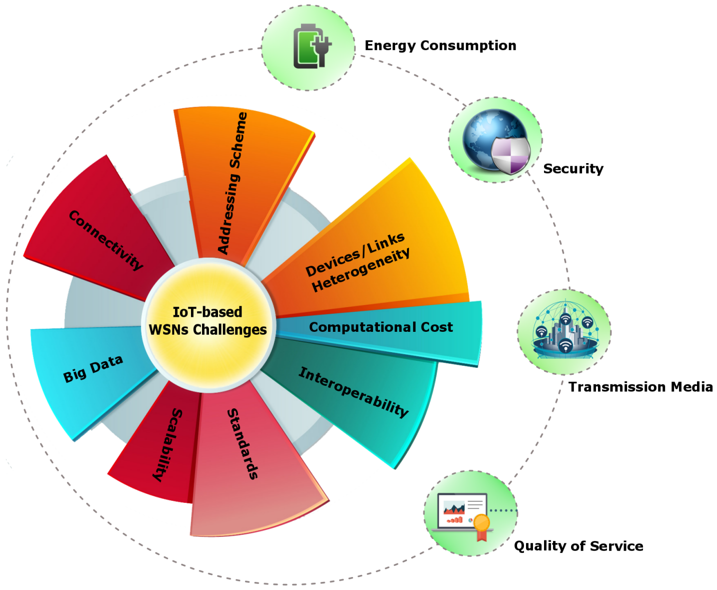


Figure 2.9 Necessary considerations for IoT-based WSNs.

2.4 Reasons and Solutions for Energy Consumption

The problem that this thesis proposes to investigate is minimising the energy consumption of sensor nodes and thus lengthening the network lifetime. Network lifetime is one of the most important metrics for the evaluation of sensor nodes but, in the literature, there are different definitions of network lifetime. Generally, it is defined as the length of time that sensor nodes would be fully operational. In other words, network lifetime is the time until the first node dies [54]. Another definition of network lifetime is the time until the first sensor node or group of sensor nodes in the network runs out of energy [61]. A node can only fulfill its mission as long as it is live, so losing a node would damage the network which would lose some of its functionalities. Hence, the main aim of any energy-efficient technique is to keep the nodes alive for longer and thus prolong network lifetime.

Various sources of energy wastage and different solutions have been mentioned in the literature and are discussed in the following sub-sections.

2.4.1 Sources of Energy Wastage

Several studies have revealed that the communication unit is relatively greedy for energy [54]. In WSNs, most of the energy is consumed in the processing, transmitting or receiving of data to fulfill the requirements of the application [62]. It is obvious that reducing data transmissions will save the energy of these constrained devices [54]. In regard to communication, a number of studies have found that a great amount of energy is dissipated in ways that make no useful contribution to the application, such as [63]:

- **Collision:** when a node receives two or more packets at the same time, a packet collision occurs [64]. Thus, the packets are either discarded or sent back to their originating node, then retransmission of these packets is required which increases packet latency and energy consumption which adversely affects the network lifetime [65].

- **Overhearing:** is a significant waste of energy, especially when node density is high and traffic load is heavy. When a node sends a packet, all nodes located within its transmission range receive the packet even if these nodes are not the intended destination [66, 67], see Figure 2.10. *Node A* wants to deliver its data to *Node B*. However, many surrounding nodes are within radio range of *Node A*. All these nodes will receive the data from *Node A*. Energy is wasted when a node receives/transmits packets that are destined for other nodes [67]. Note that *Node A* will also receive data from its surrounding nodes when they transmit their data.

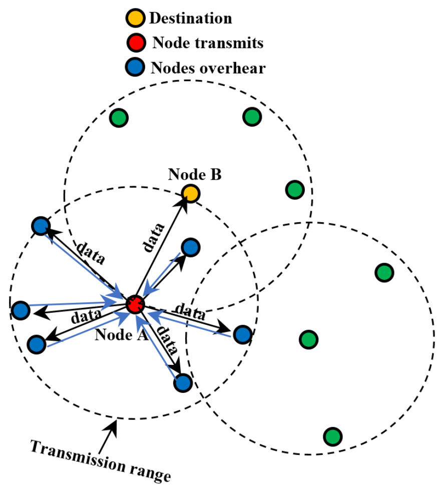


Figure 2.10 A source node transmits to its destination and neighbouring nodes overhearing the communication.

- **Control packet overhead:** is a combination of excess memory, computation time, bandwidth or other resources to perform a specific task. Thus, it is important to process the minimum number of control packets that enable the transmission [63].

- **Idle listening:** happens when a node has to stay open to an idle channel in order to receive possible traffic [63], thus a node with a higher number of neighbour nodes will be active most of the time. This is due to overhearing transmissions, neighbour nodes discovery [68] or a node may use numerous paths to deliver data to a neighbour nodes [69]. Obviously, a node with less idle listening time has better energy retention than other nodes [63].

- **Interference:** each node with two or more nodes within transmission range suffers from interference generated by the surrounding nodes. Interference increases with increase in the number of neighbouring nodes [70]. It increases both congestion and conflicting transmissions, and then retransmission may happen. Therefore, avoiding higher node interference could reduce packet loss and thus minimise the overall energy consumption of the network [71].

- **Redundant Data:** sensor nodes are generally deployed randomly which can mean that there are some regions monitored by two or more sensors at the same time [72]. However, this type of deployment will increase the reporting of redundant data in the network. As a result, energy is wasted aggregating, processing and transmitting redundant data [13]. Energy consumption could be minimised by avoiding the unnecessary operation of a node.

- **Distance:** the transmission distance (T_d) between nodes is a very important aspect of energy efficiency. The communication between a node and its CH/BS can be either single or multi-hop. Since energy consumption for transmission is proportional to the square of the distance (see Section 3.4) [73], so the power required for transmission increases rapidly with distance, which means single-hop transmission increases energy consumption if the size of the network is large. Most of the literature shows that multi-hop communication is the best way to reduce the transmission distance between nodes. Figure 2.11 shows single and multi-hop scenarios between nodes.

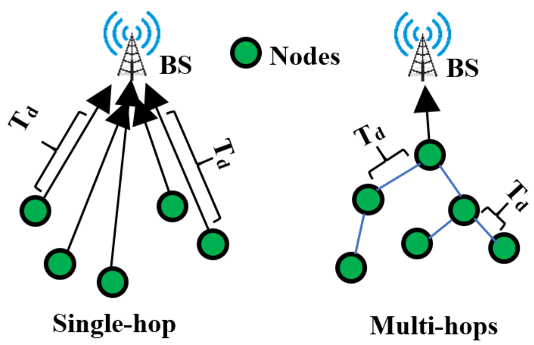


Figure 2.11 Single-hop and multi-hop scenarios.

A lower transmission distance from source node to the next-hop node/CH/BS reduces energy consumption of a node and prolongs the network lifetime [54].

- **Non-Clustering:** direct transmission distance from every source node to the BS can reduce the WSN lifetime significantly due to the additional energy consumption. As a solution, hierarchical routing protocols are adopted, see Figure 2.12 which shows chain-based, cluster-based and tree-based protocols, which are the most commonly used protocols [74]. In a chain-based protocol, nodes are arranged chain-like where one of the nodes is selected to serve as the CH node to transmit the data coming from all nodes to the BS [75]. With cluster-based, the nodes are divided into sub-clusters and each sub-cluster has some sensor nodes connected to a CH node to transmit their data to the BS [76]. In tree-based clusters, all sensing data is sent from each sensor node to its parent (CH node) based on multi-hop concept [77]. For sensor networks, clustering is the best solution for reducing communication costs and maximising network lifetime.

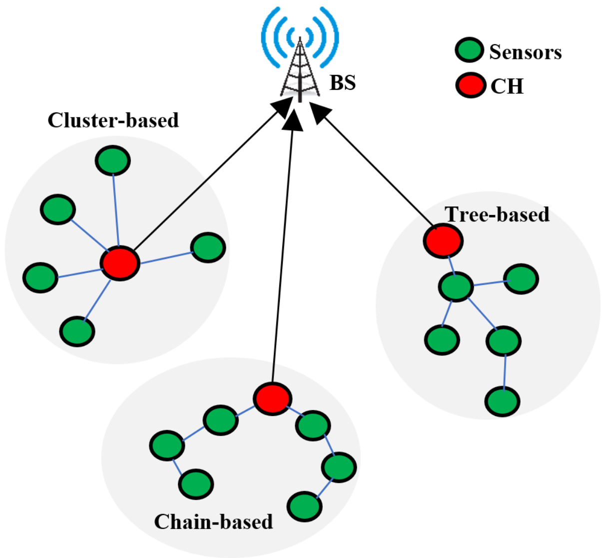


Figure 2.12 Hierarchical clustering in WSN.

2.4.2 Taxonomy of Energy Consumption Solutions

There are many methods available in the literature that can effectively reduce energy consumption and prolong the network lifetime of WSNs and IoT networks such as path selection [78], scheduling data [58], an efficient data aggregation [79], etc. These methods can broadly be classified into a number of categories as summarised in Figure 2.13 which presents a taxonomy of energy consumption solutions and techniques presented in the literature. These solutions and techniques are explained in the following sub-sections.

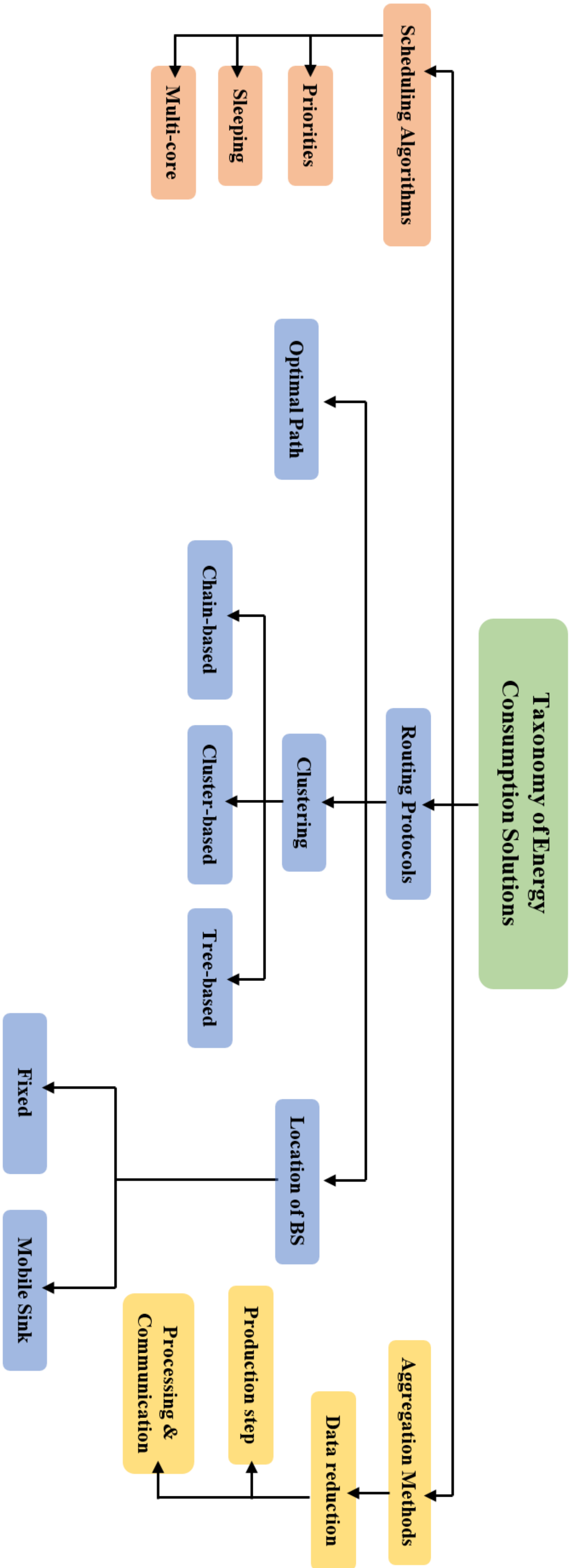


Figure 2.13 Taxonomy of energy consumption solutions.

2.4.2.1 Energy Oriented Path Selection

One of the most significant current discussions of IoT based on WSNs, is the generation of an unprecedented amount of data [36], and how to select optimal paths for transmitting such vast quantities to the final destination. Energy-efficient routing protocols are implemented to reduce the energy consumption and lengthen the network lifetime for these sensor networks. Several routing protocols for sensor networks have been proposed by various researchers to conserve energy. Most involve implementing the concept of using CH nodes within the network, selecting the optimal path from the source to the destination, or manipulating the location of the BS [80].

(i) **CH Node Selection**

Various strategies are used in the literature for CH selection to optimise energy usage. The most common three are: low energy adaptive clustering hierarchy (LEACH) [76], hybrid, energy-efficient and distributed protocol (HEED) [81] and power-efficient gathering in sensor information systems (PEGASIS) [82]. We present a brief survey of LEACH, HEED and PEGASIS in which nodes are clustered in various forms for data aggregation and communication protocols.

(a) **LEACH**

LEACH is one of the most popular strategies, in which the CH node is selected based on a probabilistic approach and the amount of energy left of the CH and the system is rotated at different time intervals [76]. Nodes that have already been the CH cannot be selected again for a number of rounds. The selected CH node broadcasts to the network and creates a schedules for each node in its cluster to transmit its data. Each node connects to the CH with a single hop and selects a random number between 0 and 1, then compares the number with a threshold value $T(n)$. A node becomes a CH in each round if the random number is less than the following threshold:

$$T(n) = \begin{cases} \frac{1}{1-P(r \mod 1/p)} & if n \in G \\ 0 & if n \notin G \end{cases} \tag{2.1}$$

where P is the desired percentage of CH, G is a set of nodes that have not been selected as CH node in the previous $1/p$ rounds. r defines the most recent round. The CH node collects the data from all the nodes connected to it, compresses the data and then forwards it to the ultimate receiver. Every node will be in standby mode except when transmitting to its CH. Figure 2.14 shows a cluster organisation for the LEACH protocol.

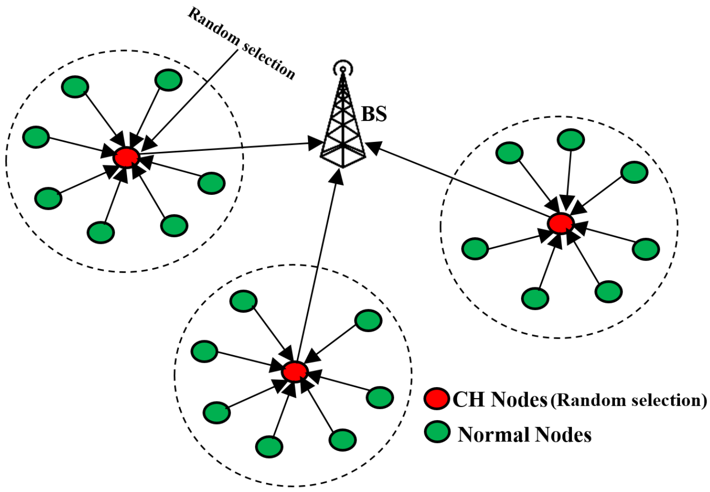


Figure 2.14 Example of LEACH protocol architecture.

A considerable amount of literature has been published on modifications of the LEACH protocol, such as LEACH-C and energy balanced-LEACH [83, 84].These studies tried to overcome the problems associated with LEACH (i.e., random process selection of CHs) and further minimise the total energy consumption for WSNs.

In the LEACH-C scheme, each node in the network can calculate its energy level and send the information about its location (possibly using GPS) to the BS. The BS uses a centralised clustering algorithm to select the CH nodes. Once the clusters and associated CH nodes are computed, then the BS selects a node with higher energy and broadcasts a message to all nodes that consist of the ID of each CH node. If the ID matches then a node is the selected CH node and its destination is the BS. Otherwise, a node has to sense and send its data to the CH node.

LEACH-C provided better clustering and longer lifetime than the LEACH protocol. However, energy balanced-LEACH (E-LEACH) improves the CH node selection by considering the residual energy of each node. Initially, each node has the same residual energy, and the CH nodes are selected randomly. From the second round, each node with the highest residual energy will become the CH node of its cluster. The E-LEACH protocol uses master cluster heads (MCH) to relay packets for those CH nodes that are far from the BS.

Similarly, Arya, et al. [85] introduced a modification of the LEACH protocol named the energy-aware multi-hop multi-path hierarchy protocol (EAMMH). This scheme introduced a new routing technique and clustering formation to deliver the data. The proposed algorithm partitions the sensing field into sub-clusters and each sub-cluster has a child-CH node. The main CH should be an optimum distance from these child-CH nodes. This means the distance between them should be balanced to reduce energy usage and thus increase network lifetime. The EAMMH scheme outperformed LEACH in terms of energy conservation by 23% but the main CH nodes can be overloaded and quickly drained of energy when surrounded by many child-CH nodes.

Cengin et al. [86] proposed the energy-aware multi-hop routing (EAMR) protocol for WSNs. The EAMR proposes fixed clusters to provide communication between the sensor nodes and the BS. In this protocol, when a sensor node is attached to a cluster, it will be a member for that cluster for the whole network lifetime. The selection of CH nodes is repeated each round, the proposed protocol allows a sensor node to act as a CH node until its energy falls below a threshold value. Sensor nodes located close to the BS forward their data direct to the BS. However, the remaining CH nodes forward their packets to the BS through intermediate nodes. The EAMR extends the network lifetime by implementing fixed clusters and reducing the number of CH node changes.

Even though LEACH and its derivative algorithms paved the way for implementing energy-efficient routing protocols, they all suffer from one fundamental problem. A node uses single-hop routing within clusters thus, it is not applicable to networks for large regions. Also, a node that is selected to become CH will die quickly if a larger area is to be supported. Because some CH nodes are located far away from the BS, the resulting large transmission distances lead to large energy consumption.

(b) **HEED**

HEED is the other common method of CH node selection. The proposed protocol overcomes the drawback of LEACH by achieving equal and uniform distribution of CH nodes in the network. In this approach, the CH node selection is based on the residual energy of each node and node proximity to its neighbours or node degree (minimum communication cost) [81]. HEED defined the average of minimum power levels (AMRP) needed by all M nodes within the cluster range, to reach the CH as:

$$AMRP = \frac{\sum_{i=1}^{M} MinPwr_i}{M} \tag{2.2}$$

Where $MinPwr_i$ is the minimum power level required by node i to communicate with the CH. Each node is assigned to only one cluster, and the node independently makes its decision based on local information to join a CH node via a single hop. Based on Equation (2.3), in HEED, every sensor hub sets the likelihood CH_{prob} of turning into a CH as:

$$CH_{prob} = C_{prob}\ x\ \frac{E_{residual}}{E_{max}} \tag{2.3}$$

Where $E_{residual}$ is the estimated residual energy in the node and E_{max} is the total energy of the node, which is typically similar for all nodes. C_{prob} is only used to limit the initial CH announcements, and has no direct impact on the final clusters.

A CH node is either a temporary CH, if its CH_{prob} is < 1, or a last CH, if its CH_{prob} has achieved 1. Analysis of the relative performance of HEED and LEACH showed that HEED improved the network lifetime by 10% [87]. Figure 2.15 shows an example of a network topology used by the HEED protocol.

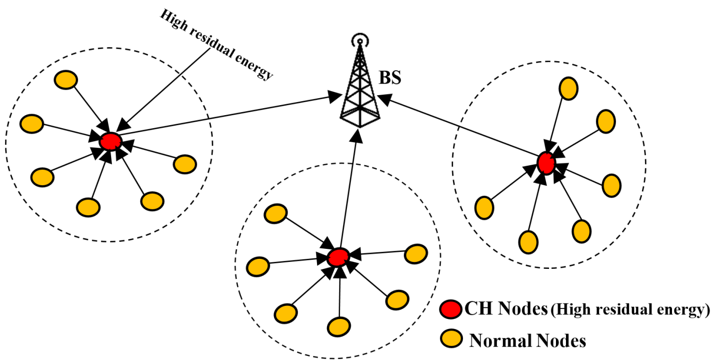


Figure 2.15 HEED protocol architecture.

A number of researchers have attempted to overcome the limitations of HEED protocol (such as more CHs are generated, the locations of the CHs, etc.) and improve its performance [88]. One example is the heterogeneous hybrid energy-efficient distributed (H-HEED) protocol. This protocol divides the sensing field into clusters and each cluster has some sensor nodes. The H-HEED protocol finds the centre of each cluster and then allocates the node nearest the cluster centre. The H-HEED protocol re-computes the cluster centres with a new assignment of nodes and allocates a node to clusters until clusters do not change for a given number of iterations. However, in this protocol, several iterations are performed to form the clusters and select a CH, this is an overhead that consumes a significant amount of energy [89]. Nevertheless, the proposed scheme increased the network lifetime of the sensor nodes by 63% [90]. Another study [91] proposed an energy-based rotated HEED (ER-HEED) protocol for WSNs. Here, the clustering formation and CH node selection are performed according to the HEED protocol. Therefore, the selection of CH node among sensor nodes in each cluster is based on the node with the highest energy. ER-HEED improves the HEED protocol by reducing the HEED cluster selection to reduce energy consumption and extend network lifetime.

In [92], a new multi-hop routing protocol was proposed, the cluster heads enhanced hybrid, energy-efficient distributed HEED protocol (E-HEED) for WSNs. The E-HEED selects the CH node based on the HEED protocol, and then grades the CH nodes according to the least transmission distance from the BS. It was claimed that the E-HEED protocol lengthened network life by 0.8 % compared to HEED.

(c) **PEGASIS**

PEGASIS is another CH node selection technique. This approach is to form a chain among the sensor nodes for the transmissions, see Figure 2.16 for the architecture of the PEGASIS routing protocol [82]. Each node receives the data from one neigbhour node and transmits it to another. Two nodes at the end of the chain forming the routing structure will send the data through the intermediate nodes to the leader node and then the leader transmits it to the BS. A leader node is randomly selected to dispatch the collected data to the BS. The main purpose of PEGASIS is to shorten the transmission distances between nodes, and thus the energy consumption of each node is minimised. However, only one node is elected as a CH node per round. It this may become a bottleneck that causes delay and retransmission of a chunk of packets. It also increases the rate of packet transmission on the node selected as a leader and thus depletes its energy quickly.

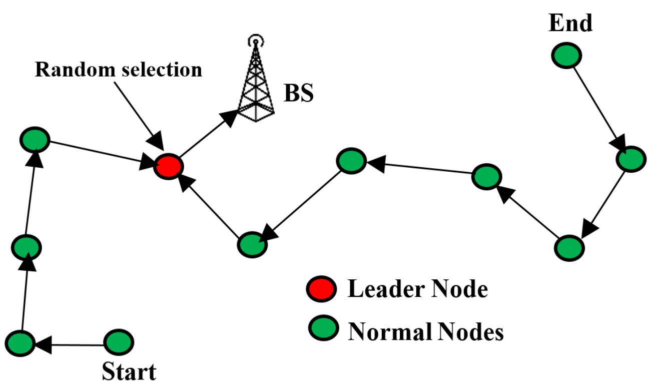


Figure 2.16 PEGASIS protocol architecture.

Table2.1 presents characteristics and comparisons of LEACH, HEED and PEGASIS based on the more important metrics:

Table 2.1 Classification and comparison of routing protocols in WSN.

Parameters	LEACH	HEED	PEGASIS	References
Type of protocol	Hierarchical	Hierarchical	Hierarchical	[93, 81]
Data delivery model	Cluster-based	Cluster-based	Chain-based	[94]
Nodes distributed	Random	Random	Random	[95]
Node mobility	Fixed	Fixed	Fixed	[94]
Multi-hop	No	Yes	No	[94]
Clustering Method	Distributed	Distributed	Centralized	[95]
CH selection	Threshold	Residual Energy	Threshold	[76, 81, 75]
Relay node	CH	CH and nodes	nodes	[76, 81, 75]
Data aggregation	Yes	Yes	No	[94]
Scalability	Low	Moderate	Low	[96]

(ii) **Optimal Path Selection**

A number of studies have considered optimal path selection for energy saving in WSNs. The shortest path approach is a commonly used methods for constructing routing trees in the many-to-one WSN [97]. The potential advantages of shortest path are minimum time delay and lowest energy consumption. Banerjee, et al. [98] proposed a heuristic algorithm based on multi-hops that perform geographical routing. This protocol selects a path with the minimum number of hops and distance from the source to the destination. The proposed scheme reduces the end-to-end node delay. In [99], authors introduced a distributed shortest path routing network from a source node to the destination. The resulting algorithm provides best link cost and maximum network lifetime.

Cota-Ruiz, et al. [100] proposed a new routing technique that is capable of estimating the distance between two non-neighbour nodes in multi-hop WSNs. This method finds all possible paths between a source node and the destination with the minimum number of hops. This leads to reducing the energy consumption and delay of the network overall. Another study [101] proposed a new centralised energy-efficient clustering algorithm for WSNs. This is the distance energy evaluated (DEE) protocol which selects the CH nodes based on the ratio between distance and residual energy of each node. The probability of being CH is determined according to the node's initial and residual energy. The DEE protocol extends the network lifetime by reducing unnecessary traffic.

Most studies have not considered the shortest path combined with balancing the load traffic in each node along the path to deliver data. A node with a higher number of neighbour nodes (within transmission range) has less energy due to overhearing, neighbour nodes discovery, or a node may be used for many paths to deliver neighbour nodes' data [69].

(*iii*) **Manipulating the Location of BS**

Several studies have proposed manipulation of BS/sink location as a means of reducing energy consumption. They found that the network lifetime can be extended by reducing the transmission distances between sensor nodes. In the work of Grossglauser, et al. [102], the idea of a mobile sink (MS) was proposed, where the sink moves in a prescribed path to collect the data within the network. In such a scheme, all nodes regardless of distance will establish a direct connection with the sink. Therefore, the total link length of the network will be very high, especially when a node is located on the border of the network, consuming more energy than other nodes which are close to the sink. The optimal location for a mobile sink (OLMS) for WSNs is suggested by [103]. In this approach, clustering is performed and CH nodes are selected at each round. The proposed protocol determines the best location of the MS based on the minimum energy cost for data delivery of CHs and thus reduces energy consumption and lengthens the network's lifetime.

In [54], the authors also examined a tree-based mobile sink (TBMS) technique. It adopts the sorting algorithm and the multi-hop concept to create the routing structure. The proposed method introduces a MS that gathers the data from the sensing field but in a way that reduces the hop distances and thereby extends the lifetime. However, authors assume that the MS moves randomly in the sensing field. Therefore, there is no guarantee that the MS will cover all the sensing field, or it might take too long when the sensing field are extended. Of course, if the speed of at which MS moves is too fast or slow, then it can cause high packets loss and/or more delay.

However, some important factors such as interference effects and dynamic network topology should be considered when designing wireless sensors routing protocols. This is because of the challenges that may arise as a result of the characteristics of the environment in which these networks are deployed [71, 104].

- **Interference Effect**

  High node density in the sensing field, can lead to interference effects which can adversely affect energy consumption in sensor networks. According to an investigation by [71], interference occurs during transmission and can cause packet loss. In such a case, lost packets need to be retransmitted and every retransmission is energy wasted [65]. Thus, these authors suggested avoiding paths with higher interference levels [105]. In [106], the authors proposed a new routing metric that selects a path with minimum interference of transmitted data. The proposed algorithm balances the traffic load and significantly reduces congestion in the network. An energy-aware interference-sensitive geographic routing (EIGR) was investigated by [107]. The EIGR adaptively uses an anchor list to guide data delivery and selects the minimum interference path from the energy-optimal relay region for data delivery. The EIGR adjusts the transmission power which is only required to deliver the information to the forwarding node. The proposed protocol focuses on reducing interference and minimising the total energy consumption of the network.

Other researchers [108] have addressed the problem of interference in WSNs, and here the proposed scheme finds the shortest path from source to the destination which avoids interference areas based on an ad-hoc, on-demand distance vector (AODV) protocol. Liu, et al. [109] introduced a full-duplex BackCom network, where a novel time-hopping spread-spectrum (TH-SS) based multiple access scheme was implemented. The proposed protocol enabled simultaneous forward/backward information transfer from one device to another. The interference in such networks is suppressed by the proposed multiple-access scheme based on the TH-SS technique and allows wireless energy harvesting from interference.

However, these strategies did not consider the interference caused by neighbouring nodes of the next-hop node. Increasing the number of neighbour nodes adjacent to each node (within transmission range) generates an increase in interference [110]. As a result, increasing the packets loss and decreasing the network lifetime.

- **Dynamic Network Topology**

  With multi-hopping, sensor nodes depend on some intermediate nodes to deliver their packets to the final destination. Some of these intermediate nodes may be blocked or fail due to exposure to harsh environment, interference, physical damage or lack of power during transmitting and receiving packets [111]. The probability of sensor node failure increases with the rise in the number of sensors and sensing field. A sensor node is reported as a failure node when a node cannot receive or communicate messages with neighbour nodes for more than a specific period of time and thus excluded from the routing path. Such node failure should not affect the overall task of the sensor network [112]. WSN routing protocols should have the ability to recover from the failure of a sensor node [104]. Therefore, a routing protocol must follow and link with new nodes dynamically to deliver the data gathered by other nodes to the intended destination. For example, Figure 2.17 clearly shows that source1 sends its data to the destination via intermediate nodes. Unfortunately, *path*1 and 2 failed to carry the data to the final destination due to malfunctioning of some nodes. Therefore, a new path is required to deliver the packets to the final destination (i.e.,path3).

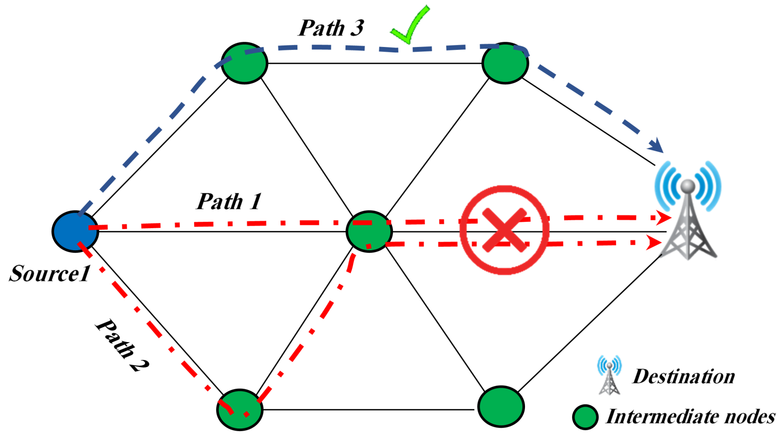


Figure 2.17 Recovery from node failure. *Paths*1 and 2 failed to carry the data from source1 to the final destination. Therefore, source1 establishes another path (*path*3) to deliver its packets.

Several studies have been carried out to provide routing protocols that help to recover from a failed node in the network. Most of this research focused on providing a backup node or finding an alternative node to avoid link failure from source to destination. In [111], authors reported a mobile sensor node acting in cooperation with a static node in order to fill gaps created by faulty nodes in the sensing field, which resulted in overcoming the failure issue and increasing the network lifetime. Other work [113] proposed a new procedure that could replace the dead CH node with backup cluster heads (BCH) in the case of CH node failure. One study [114] has proposed an energy-efficient recovery and backup node selection for IoT systems. The system includes backup nodes which are in sleep mode until required due to a node failing and then are enlivened. This results in energy-efficient solution and maximises the network lifetime.

In another study [115], the authors proposed a new algorithm to create primary and alternative paths in the network. The proposed method reroutes the traffic from nodes connected directly to the failure node, and reroutes the traffic in an alternative path. In addition to [115], [116] suggested a novel path redundancy-based algorithm which called for dual separate paths (DSP). The DSP algorithm provides fault-tolerant communication for WSN applications. This protocol implements two separate paths between a source and destination and thus improves the network traffic performance. The cluster-head recovery algorithm (CHRA) [117] uses a check-pointing techniques to create a recovery path within each cluster. In case of CH failure, a recovery path is created for sensor nodes connected to the failed CH node.

2.4.2.2 Energy Oriented Message Scheduling

Sensor nodes around a CH node naturally create a many-to-one traffic pattern [77, 118]. Congestion generally occurs when the traffic load on a particular node exceeds the available buffer capacity which leads to successfully delivery for only some of the packets and thus packet retransmission is required [119]. Retransmission of data will, of course, consume additional energy. Most previous studies do not consider packet overheads due to the retransmission of packets. For example, when a connection-oriented protocol such as the transmission control protocol (TCP) [120] is established, for reliability it uses three-way handshakes to establish the connection between the source and destination. This leads to a significant increase in network traffic and thus increases the data transmission rate and volume. Therefore, during the implementation of the scheduling techniques, packet overhead must be considered since retransmission adds a burden on the network and reduces network lifetime. To minimise power and memory usage, superior scheduling protocols that consider packet overhead are required.

Various scheduling algorithms have been proposed for use in sensor networks. A scheduling method for nodes that are located between two or more clusters (border nodes) is introduced in [58]. A switching technique between listening and sleeping modes is adopted by the nodes in this scheduling approach. However, a node that is located between more than two clusters, will often switch to the listening mode and this will lead to the unnecessary consumption of energy which decreases the network's lifetime. The proposed protocol introduces a unified scheduling method to solve the problem of diversified scheduling of border nodes.

Another study by Gupta and Rao [121] proposed demand-based coverage and a connectivity routing protocol to provide the desired coverage and meet connectivity requirements in WSNs. The idea was to use a probabilistic approach to calculate and reduce the sensing range of the sensors. It also uses a sleep scheduling protocol to switch ON/OFF the communication radio which results in saving energy. Abdullah and Yang [114] proposed clustering IoT networks into sub-groups and placing within each group, a broker (CH node). The broker was deployed to gather information from the sensors around it and forward the data to the final destination. The short process time (SPT) algorithm was implemented at the broker level to select and deliver packets based on their arrival time. Each message is presented as Mess(R_{time}, T_{trans}), where R_{time}, T_{trans} are request time and successful transmission time periods, respectively. The SPT algorithm was applied when the network was unstable (traffic intensity > 1). The proposed method promoted IoT system efficiency by improving service response time and reducing the overall energy consumption.

The earliest deadline first (EDF) scheduling algorithm has also been used to manage real-time tasks and to place processes in a priority queue. High priority is assigned for packets closest to their deadline or expiry in the queue [122–124]. Houben, et al. [124] have discussed reducing energy consumption in real-time systems by sorting the tasks with enhanced EDF to vary the processor modes determined by supply voltage, frequency and performance requirements. However, several request packets in the queue can have the same deadline, and the EDF algorithm does not consider the time redundancy management of these packets, nor does it give priority to packets that come from longer distances, with more hops which leads to higher energy consumption. Gomathi and Mahendran [125] implemented the nearest job next (NJN) scheduling protocol. The NJN algorithm always selects the nearest sensor node as the first job to collect and delivers its data to the ultimate receiver, which helps the system to reduce latency.

An approach suggested by [126] proposed an energy-efficient heterogeneous dual-core processor for IoT devices. The proposed scheme included an ultra-low-power CoreL and a fast CoreH processor. This technique schedules the tasks between these two processors and runs multi-tasks at the same time. However, the problem with the multi-core processor system is that requires a large memory to hold the packets during processing. Additionally, overheating due to the use of two processors can cause the damage to the device [127].

With the increase in node density, scheduling different types of data packets such as high or low priority data at CH nodes is essential for reducing energy consumption, capacity and end-to-end delay. For example, if the queue is scheduled inefficiently then packet drop might happen and retransmission is required. This problem can be serious for border nodes which use a number of hops and access many devices to reach the required destination. This means that these nodes consume more energy than other nodes in the network located close to the destination. Therefore, scheduling at CH nodes which gives priority to the data that comes from further nodes is crucial for energy saving.

2.4.2.3 Efficient Data Aggregation

The integration of WSNs and IoT elements is all about connecting devices to the internet, making them more convenient to use and maximising their efficiency [128]. Such integration can generate a large amount of data by these devices. There is considerable data redundancy in such networks due to dense deployment. Redundant data requires a considerable amount of energy to process and transmit [129]. Since each sensor is provided with only limited power eliminating data redundancy would considerably improve energy consumption overall for IoT networks.

A number of studies have focused on reducing the number of data transmissions and data volume for sensor networks. A clustering algorithm is proposed in [130, 72] to reduce energy consumption and prolong the network lifetime. The proposed scheme divides the sensing field into cells and each cell selects one node to act as a cell head for all of them. Thus, one node sends all data collected to the cell head node, which accepts the data from the connected nodes, removes redundant data, and then the remaining data are delivered to the final destination.

In [129], an energy efficient in-network RFID data filtering scheme (EIFS) was proposed. The algorithm divided the sensing field into sub-clusters and each cluster has a CH node. The CH node removes duplicated data from its member nodes and forwards the filtered data to the final destination. In another study [131], the authors proposed a method that reduced the number of data transmissions, whereby the proposed method controlled the RF-transmit operation. ON/OFF begins only when the data sensed was largely different from the previous state. In other words, the RF did not send data to the ultimate receiver if the current value is approximately equal to the last recorded value.

Recent studies have confirmed that cloud computing technology offers several advantages to WSNs and IoT in terms of scalability, storage, computing tasks, etc. over the internet. It can be applied to analyse the data generated by sensors and IoT devices [38]. For example, Vincent, et al. [38] investigated a cloud-based architecture to enable data gathering, monitoring and processing for IoT devices. The proposed system collects the data from different IoT devices and sends it to the end-users via cloud infrastructure. This study aimed to provide inter-operability and an efficient communication mechanism for IoT devices. The study by [132] implemented a real-time monitoring system for soil nutrient using WSN. In the proposed system, sensors measured the macro-nutrient of soil and transmitted the data to the cloud. The user can access these data and monitor the field conditions from anywhere via a website.

Several studies have highlighted the need for real-time monitoring that can gather the data from IoT devices based on the message queuing telemetry transport (MQTT) protocol. MQTT is widely used in IoT due to its low-overhead protocol that emphasises the bandwidth and processor limitations of the IoT devices. It uses publish/subscribe pattern and translates messages between sensors, devices, servers and applications [3]. MQTT with IoT has been used in many applications such as environmental monitoring, health-care, and industry. In [133], the authors proposed a network of IoT monitoring devices for fire detection. The proposed system was able to detect and monitor fires and send the information to the concerned people and authorities so that preventive measures could be taken. In [134], the authors used sensors to measure body temperature, pulse rate, body movement of patients and this measured data was uploaded to the MQTT server. The proposed system aimed to help the doctors to monitor patients from any location and at anytime. It also helped patients to view and check their health condition remotely. Another study by [135] proposed a web-based interface for monitoring motion and controlling a arm robot. The proposed system gives low latency data transmission via using MQTT protocol.

2.5 Research Gaps

The literature review demonstrated the existence of many avenues for reducing energy consumption and lengthening the network lifetime. It presented many possibilities and also identified numerous limitations. This chapter has scrutinised the literature to obtain an insight into the perspectives addressed by previous researchers and gaps left by existing solutions.

- **Limitation of Scheduling and Routing Techniques**

  In the sensing field, sensor nodes depend on other intermediate nodes to deliver their packets. Some sensor nodes send their data over single- or multiple-hops to reach the BS through CH nodes. The CH node can become overloaded due to the number of nodes connected to it. The probability of retransmitting some of these packets at the CH nodes will increase. Therefore, data sent by the sensor nodes must be prioritised at CH nodes based on the energy consumed by each packet. However, to the best of the author's knowledge, no report has been found assigning high priority to packets that come from the furthest distances to the CH nodes. Since these packets more quickly exhaust the network resources because they require more links and nodes to reach their ultimate destination, this research suggests the introduction of a novel scheduling algorithm called the long hop (LH) to optimise energy usage in sensor networks. The LH algorithm will give high priority to the scheduling of packets arriving after a greater number of hops, from longer distances. These will be served first at the CH nodes to prevent them being retransmitted, and so conserving energy.

  The literature review also introduced the energy consumed due to delivering data from the source node to the destination. Based on the multi-hop concept, the transmitted packets access multiple nodes and experience a greater number of hops to reach the final destination. Each node involved in the transmission process has a different number of forwarding nodes. However, a higher number of forwarding nodes causes higher energy consumption. Thus, we propose a new routing technique that sends data to the next-hop node within a shorter transmission distance and fewer forwarding nodes to optimise the energy consumption. The proposed protocol avoids forwarding data to nodes that have many forwarding nodes, thereby balancing the load traffic and improving the network performance and lifetime.

Although a large and growing body of literature reports attempts to reduce the energy consumption of sensor networks, this author believes that there are only a few research studies on theoretical energy analysis of sensor nodes based on less transmission distance, interference and the creation of CH nodes. Interference is also one of the main factors that cause data collisions and consequent energy wastage. A node with fewer neighbours has less overhearing and interference. Therefore, we propose a new routing technique that selects the next-hop node to be one with least transmission distance and fewer neighbour nodes and thus less interference. The proposed scheme also introduces a new method that selects CH nodes around a single BS based on lower transmission distances. Both techniques minimise energy consumption and lengthen the network lifetime.

- **Limitation of Aggregation and Transmission Techniques**

  As shown in the literature, previous research focused on efficient transmission data in the WSNs and IoT applications. However, most of these studies considered either eliminating redundant data or monitoring active devices remotely. Thus, this research proposes investigating both the filtering of redundant data and the remote monitoring of the behaviour and condition of the sensor nodes. In this proposed technique, the IoT device filters out the duplicated data before sending it to the end-user via a cloud infrastructure. The end-user, however, predicts the duplicated data which was eliminated by IoT device and then stores it and plots it in real-time communication. In addition to this, the end-user assesses the entire system by monitoring the status of the IoT application. This work aims to reduce the volume of data transmissions which, in turn, will reduce the process of scheduling and routing data on each device on the network, as a result reduces the energy consumption and success the IoT technology.

2.6 Research Methodology

The main concern of this thesis is the energy consumption of WSN-enabled IoT networks. Energy is definitely one of the most important issues of these networks. The sensing field may require a number of sensor nodes to sense the data and transmit it to the ultimate receiver. However, these nodes are restricted by limited battery power and the battery life is a major constraint on the sensing field. Therefore, the thesis proposes different methods that can reduce energy consumption and extend the network lifetime.

In order to achieve the proposed methods, we need some minimum software specifications and necessary simulators to simulate the results which are discussed below.

- It is assumed that sensor nodes are employed randomly in a large geographic area to collect the data from the environment and transmit it to the ultimate receiver.

The sensing field is divided into sub-clusters and each cluster has a different number of sensor nodes and a single CH node. Each sensor node can monitor the environment and directly send the data to the CH node with a single hop or via intermediate nodes. Sensor nodes forward the data to the next-hop node with least transmission distance and least number of neighbour nodes. The CH node then compresses the data and forwards them to the BS. The selection of the CH node is based on the shortest transmission distance to the BS. The CH nodes may be overloaded, and the probability of packet drops increases, with consequent retransmission of packets. Thus, we propose a new method that assigns high priority to packets that consume more energy so they are served first at the CH nodes in order to reduce the energy consumed by the retransmission process. The BS acts as a gateway between the sensor nodes and the end-user. Matlab software version R2015a uses to demonstrate the performance of the given network and evaluate the results. We use different numbers of nodes and sensing areas for simulating the WSNs. The proposed model depends on some mathematical equations and implements in the Matlab environment in order to calculate energy consumption and the network lifetime.

- Aggregation and transmission of data are considered as the main reasons for energy consumption by WSNs and IoT devices. In order to reduce energy consumption, the hardware presents means to eliminate redundant data and consequently reduce the number of data transmissions. A distributed software and hardware system are developed that can sense the analogue data via six sensors (pressure transmitter (Type 663)) from the blade of the gas turbine engine and forward it to a Raspberry Pi 3 model B (RasPi) through a Custard Pi 3A (CPi). The sensors are interfaced and powered by the CPi. They are placed in different positions in the test section to sense and forward the data to the RasPi via the CPi. The CPi is linked to and powered by the RasPi. The RasPi is used to process and forward the data to the monitoring server via the cloud infrastructure. Raspbian is used as the operating system for the RasPi. However, Windows 10 is used as the operating system for the monitoring server. Python language version 2.7 is installed in both devices; RasPi and the monitoring server which helps to read the sensor values, process and forward the data to the monitoring server. It also assists in receiving, plotting and storing these data sets on the monitoring server database. The study also provides a remote monitoring system for the end-user that can check and track the performance of these sensors/IoT devices during real-time communication.

2.7 Summary

WSNs play an important role in IoT technology which is necessary to provide sensing services and to monitor environmental conditions. Most of these devices run on batteries or by energy harvesting, which has a limited shelf life. Energy-saving becomes one of the most essential aspects of WSNs and IoT networks to lengthen network lifetime. Therefore, this chapter first explained the main components of sensor nodes and how WSNs work. Then, the chapter focused on energy consumption and explains the main reasons for energy waste in WSNs. The chapter critically reviews related literature on energy efficiency solutions in sensor-enabled wireless network environments. Gaps in the research are identified and discussed.

Following these, the co-operation between energy-oriented path selection and a message scheduling framework for a sensor-enabled wireless network environment is presented in the next chapter. The proposed technique reduces the overall energy consumption and prolongs the working life of the network.

Chapter 3

Energy Oriented Path and Message Scheduling

Routing and scheduling algorithms play a very important role in terms of energy efficiency for the wireless sensor networks (WSNs) that enable the Internet of Things. Suitable techniques can reduce energy consumption and extend network lifetime. In this chapter, a new routing algorithm is proposed that forwards the data to the next-hop node with least transmission distance and least number of forwarding nodes. The chapter also introduces a new scheduling algorithm that assigns high priority to packets that consume more energy so they are served first at the cluster head nodes in order to reduce the energy consumed by the retransmission process.

3.1 Introduction

SENSOR nodes are usually configured in a mesh network and send their data to the base station (BS) through intermediate nodes and cluster head (CH) nodes [83]. The intermediate nodes are responsible for sensing the nearby environment and forwarding packets received from other nodes towards the BS. These nodes consume most energy during the receiving, processing and forwarding of packets from other nodes [54]. However, these nodes are restricted by limited battery power, processing ability, memory, and radio communication [85]. Intermediate nodes that receive and forward data from a greater number of forwarding nodes consume more energy than those nodes that receive and forward data from fewer forwarding nodes (see Figure 3.1). *Nodes A* and *B* have a different number of forwarding nodes associated with them. However, *Node B* has a higher number of forwarding nodes. This means that *Node B* consumes more energy than *Node A* receiving, processing and transmitting packets from its forwarding nodes. To minimise energy consumption, a suitable routing strategy needs to be adopted, one that takes into account the number of forwarding nodes associated with each node on the path.

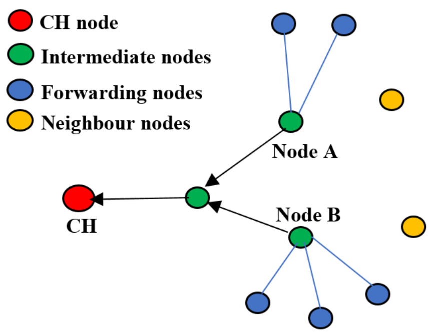


Figure 3.1 Intermediate nodes, forwarding nodes and neighbour nodes.

In WSNs, the sensing field is divided into sub-clusters and each cluster has some sensor nodes and a CH node. The sensor nodes will be located at different positions, some further, some closer to the CH. Some of these nodes send their packets to the CH node in a single-hop over a short distance. However, some will send their packets over a longer distance and may access many intermediate nodes in order to reach the CH node.

When a particular event occurs in the sensing field, sensor nodes transmit a large volume of data to the CH node through intermediate nodes. Thus, the CH node may be overloaded and the probability of packet drops increases, with consequent retransmission of packets [136]. This process will depend on a number of factors such as packet arrival rate, the timeout for message expiry, or simply due to the limitations of the node because of its constrained nature (low processing ability, memory and energy resources) [137, 138]. Also, if the transmission control protocol/internet protocol

(TCP/IP) is considered as a connection-oriented protocol in WSNs, that means the sensor nodes deliver their data in three phases: Request-to-send (RTS), Clear-to-send (CTS), and then establishing the connection between the source and destination [120] (see Figure 3.2). This TCP connection requires an extra overhead in the WSN, especially for the small amount of data transferred in this network.

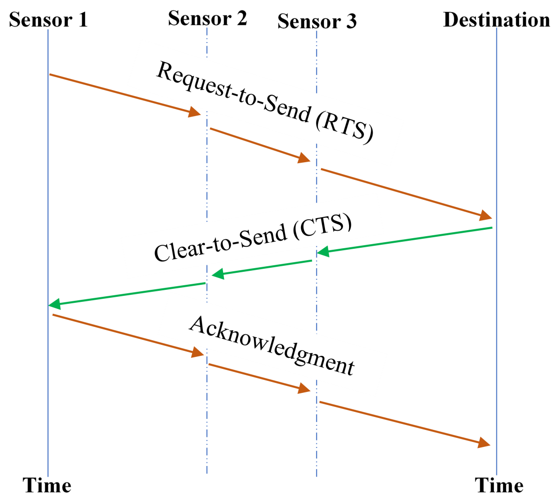


Figure 3.2 Data transaction and aggregation over TCP/IP protocol.

In the case of packet retransmissions, each retransmission requires a TCP header together with the packet. However, the retransmission packets of further nodes require to travel through all the intermediate nodes to reach the target. This can significantly impact the system performance particularly if the retransmission is not optimised correctly. It adds additional load to the already constrained network and contributes to the power depletion of the entire network. In order to minimise energy consumption and extend network life, the priority at the CH nodes should be given to those packets that consume more energy during the transmitting, receiving and processing.

This chapter proposes an energy-oriented path selection and message scheduling framework for sensor-enabled wireless network environments. The energy-oriented path selection scheme differentiates between nodes on the path towards the CH and chooses the node that has fewer forwarding nodes to be the next-hop node. The chapter also proposes a new scheduling algorithm called long hop (LH) that operates at the CH nodes. The LH algorithm assigns high priority for packets coming with more hops and further distances to be routed first to the final destination in order to prevent these packets being retransmitted. The proposed schemes balance the load on each node in the network and minimises the total number of packet retransmissions and, therefore, reduces the overall energy used in the network.

3.2 System Model

We consider a system composed of a number of sensor nodes which are deployed randomly in a large geographic area to monitor and record the physical conditions of the environment. These nodes are connected in a mesh network where the distances ($d_{i,j}$) between any two nodes is based on Euclidean geometry [139]:

$$d_{i,j} = \sqrt{((x_i - x_j) + (y_i - y_j))^2}, i, j = 1, 2, 3, ..., N \tag{3.1}$$

Where N is the number of sensor nodes in the system model, and (x_i, y_i) is the location of each node in the given sensing field.

The sensing field is divided into subgroups using tree-based clustering, and each group has a different number of sensor nodes that connect to the CH node. The communication between the nodes in each cluster is based on a tree routing structure. All nodes have the same communication range and initial energy level. Each sensor nodes collects data from the environment and sends it to its CH node with a single hop or via intermediate nodes. The CH node then compresses the data and forwards them to the BS, as in most WSN-assisted IoT applications. The BS is fully powered and placed in the centre of the sensing field and acts as a gateway between the sensor nodes and the end-user, see Figure 3.3. Matlab simulation was used to demonstrate the performance of the given network and Tables 3.1 [86], 3.2 [92] and 3.3 [54] show the initial parameters of the WSNs for simulating different numbers of nodes and sensing areas based on Table 3.4 scenarios.

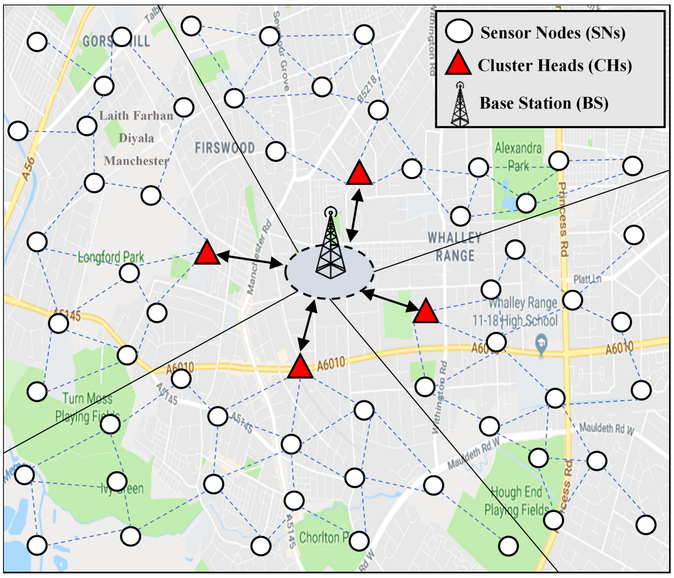


Figure 3.3 Typical WSN system architecture.

3.3 Proposed Power Saving Scheme

The aim of the proposed power-saving scheme is to balance the load on each node on the transmission path and optimise the retransmission packets to maximise network lifetime. The proposed scheme is divided into two main methods which are: shortest path fewest links (SPFL) algorithm and long hop (LH) scheduling algorithm. The number of sensor nodes, sensing field and other parameters for both schemes is based on the parameters shown in Tables 3.1 [86], 3.2 [92] and 3.3 [54], and scenarios shown in Table 3.4, see below.

3.3.1 Shortest Path Fewest Links (SPFL) Protocol

The aim of the proposed scheme is to reduce transmission distances and balance the power usage among the nodes, and thus increase the network lifetime of WSNs. The proposed scheme assumes that all sensor nodes have the same capabilities such as energy, processing, memory, global positioning system (GPS) tracker, etc. [140]. The connections of the tree routing structure are based on the distance between the nodes and their BS, and the number of forwarding nodes to each node on the path. The operation of the proposed scheme is composed of two major phases: set-up phase and steady-state phase.

- **Set-up phase:** after the sensor nodes are deployed in the sensing field and assigned with the unique ID numbers. The BS sends hello message to discover all nodes that are located in the sensing field. Sensor nodes receive the BS message and send a response message back to the BS. The message contains a unique ID number and positional information of each node. The BS gets the sensor nodes information and broadcasts the routing information table to all nodes in the sensing field that are related to it and then, the tree-based clustering is created. Based on the received information from the BS, each node knows the length of the path and the number of hops to the BS through the assigned CH node. Based on these steps, the nodes and BS can communicate and share the data with each other, as shown in Algorithm 3.1, lines 1-11.

- **Steady-state phase:** the performance of any transmission is negatively affected by the transmission distance. Therefore, the shorter the transmission distance, the lower the energy consumption. In this respect, the proposed scheme selects the next-hop node according to the shortest transmission distance. In the case of a node with two neighbour nodes at the same distance in the direction of the BS, the SPFL selects the next-hop node which has fewest number of forwarding nodes, see lines (15-22) in Algorithm 3.1.

A brief example of the proposed algorithm incorporating the above steps is shown in Figure 3.4. Each node is connected directly to the neighbour nodes located within the transmission range. The tree-cluster routing structure is established to enable the communication between the nodes and their BS. *Node A* forwards its data to the *BS* through the intermediate nodes. The packet follows the shortest transmission distance to connect with the next-hop node. However, *Node C* is located on the path, it has two next-hop nodes with the same transmission distances to which it could deliver the *Node A* packet. In this case, *Node C* takes the decision based on the SPFL policy, to transmit to the node that has fewest forwarding nodes connected to it as indicated by the red arrow. The proposed protocol aims to reduce the transmission distance and avoid forwarding data to a node that is overloaded with many forwarding nodes. This means that the proposed protocol balances the load traffic between the nodes in the sensing field and improves the network lifetime. Figure 3.5 shows the proposed algorithm with the aid of a flowchart.

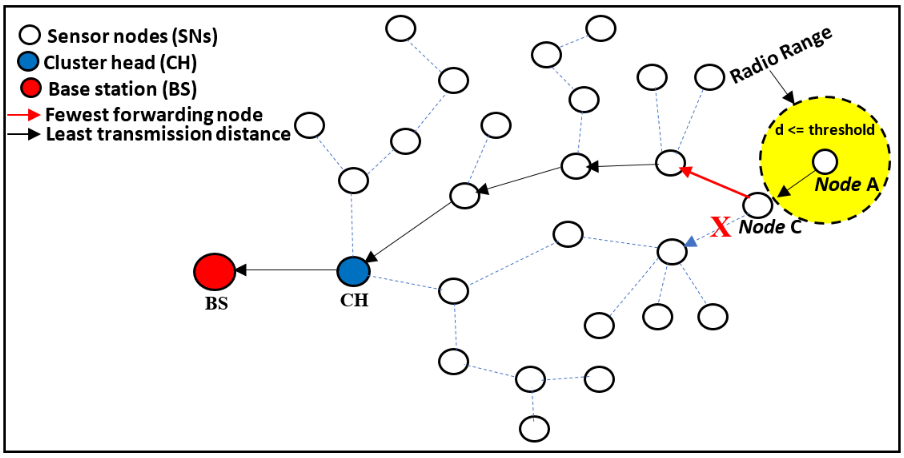


Figure 3.4 Routing structure of SPFL.

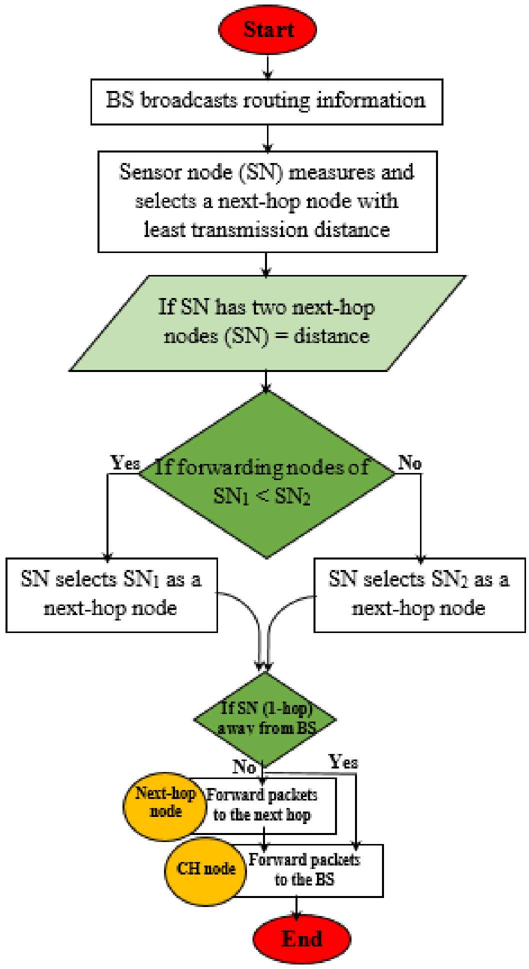


Figure 3.5 Flow-chart of SPFL algorithm.

Algorithm 3.1 : Pseudo-code of SPFL algorithm.

```
Set N = number of sensor nodes
Input: = N, M, locations of N, number of forwarding nodes
Output: A tree-cluster routing structure
procedure BROADCASTING ROUTING INFORMATION
    BS broadcasts routing information to the N
    for all N do
        (i= 1; i <= N; i++)
        Calculate d(i,BS) and number of hops
        if N ∈ network then
            N get routing information from the BS
        else
            N out of coverage area (in sleep mode)
        end if
    end for
end procedure
procedure GEOROUTINGSPFL
    for all N ∈ neighbours do
        if distance_(i) ≤ threshold then
            Send to target node
            if (N) has two minimum distances equal and linked
              with two different next-hop nodes then
                if forwarding nodes of N_1 < N_2 then
                    Select N_1 as the next-hop node
                end if
            end if
        end if
    end for
    Send packet to the target node
end procedure
```

3.3.2 Long Hop (LH) Scheduling Strategy

In the sensing field, sensor nodes rely on intermediate nodes to communicate with other nodes located out of transmission range. As previously mentioned, some of the sensor nodes send their packets with a single hop, however, some of them send their packets with multiple intermediate nodes to reach the target. By considering the energy consumed in each node on a path during the transmitting, processing and receiving of a packet, it is obvious that packets coming from further away, that access a higher number of intermediate nodes before reaching the BS, consume more energy and memory than packets from close by, which access fewer intermediate nodes.

Figure 3.6 gives an example of energy consumption for a different number of nodes in a sensing field, each node was located in a different location and used a different number of intermediate nodes to forward its packet to the BS. The figure was generated randomly by picking sixteen nodes from the sensing field, based on parameters listed in Table 3.3 and Table 3.4 (scenario 4), On page 78. The total energy consumption for each node was calculated for the first transmission only. It is equal to the summation of energy consumed in the transmitting, processing and receiving data in each intermediate node on the path. The calculation is done based on Equations 3.7, 3.8 and 3.13.

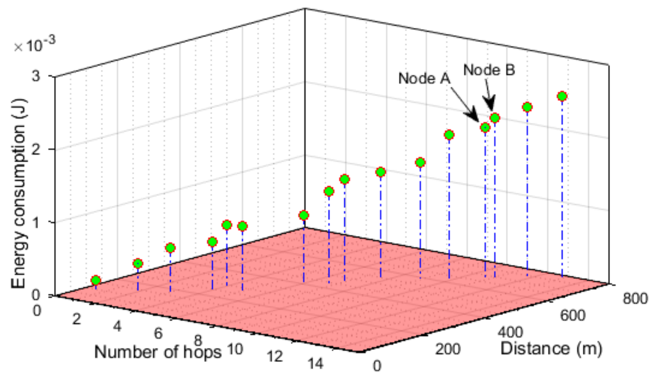


Figure 3.6 Energy consumption vs number of hops and distances.

It appears from the figure, generally, that data which accesses a higher number of intermediate nodes and travels a greater distance consumes more energy than data travelling a shorter distance and requiring fewer intermediate nodes. For example, the energy consumption for data requiring 14 hops and travelling 800 meters (0.0028 joule), is higher than for data with 3 hops and travelling 250 meters (0.00042 joule). It is obvious from Figure 3.6 that *Nodes A* and *B* use the same number of nodes to forward their data (i.e., 11 nodes). However, *Node B* consumes more energy (0.0018 joule) than *Node A* (0.0015 joule) as it is located further from the BS. This is because the power required for transmission increases rapidly with distance (see Equation 3.7). We conclude that to reduce energy consumption, it is crucial, at overloaded nodes, to give higher priority to packets arriving with more hops from a greater distance to prevent retransmission of these packets and so extend network lifetime.

Thus, the LH algorithm is implemented at the CH nodes to give higher priority to the packets that consume more energy during the transmitting and receiving process in order to prevent these packets from retransmission. However, the proposed algorithm is divided into two main steps which are: the activation process and the selection and forwarding process.

– **The Activation Process**

Due to the limited resources of the CH nodes, they cannot process all of the arriving packets simultaneously and they can become overloaded. The packets line up in a queue and wait until being transmitted to the final destination. The higher the packet arrival rate (queue length), the greater the packet discard rate will be [141].

We use M/M/1 queue to model the arrival and service rate for all packets arriving at each CH node, where the arrival rate of data packets is λ packets/s and is determined by a Poisson distribution, and service rate is μ packets/s which has an exponential distribution [142, 143]. We calculate the traffic intensity (P) of these packets as given in the equations below [142, 143]:

$$\lambda = \frac{1}{T_{request}} \tag{3.2}$$

$$\mu = \frac{1}{T_{trans}} \tag{3.3}$$

$$P = \frac{\lambda}{\mu} \tag{3.4}$$

$$P = \frac{T_{trans}}{T_{request}} \tag{3.5}$$

Each packet has a successful time of request T_{trans} and request time period $T_{request}$ [142]. Based on these equations, the total traffic intensity for the overall system in each CH node can be determined as [142]:

$$P = \sum_{1}^{n} \frac{\lambda}{\mu} = \sum_{1}^{n} \frac{T_{trans}}{T_{request}} < 1 \tag{3.6}$$

Where $0 \leq$ P < 1, the queue performs reasonably and the CH node can handle more packets. However, if P is ≥ 1, this means that the CH node is busy and the arrival rate equals or exceeds the transmission bit rate. Thus, the probability of packet drop increases, caused by queue overflow [144]. In such a case, the LH algorithm is activated and the scheduling policy starts to rearrange the forwarding of the packets.

– **The Selection and Forwarding Process**

Based on the SPFL scheme, each CH node knows the length and the number of hops for every node that is assigned to it. In order to reduce the energy consumption, the CH node selects the packets that come with a higher number of hops to be the first forwarded packets to the BS. These packets consume more energy due to their reception, processing and transmission at each node along their path.

When two or more packets in the queue have been subject to the same number of hops, the proposed algorithm takes into consideration a second parameter, the distance travelled, to select a packet for forwarding. The packet coming from the furthest distance is served and forwarded first to the BS.

These steps are presented in Algorithm 3.2 in lines 12-17. Figure 3.7 introduces the proposed algorithm with the aid of a flowchart.

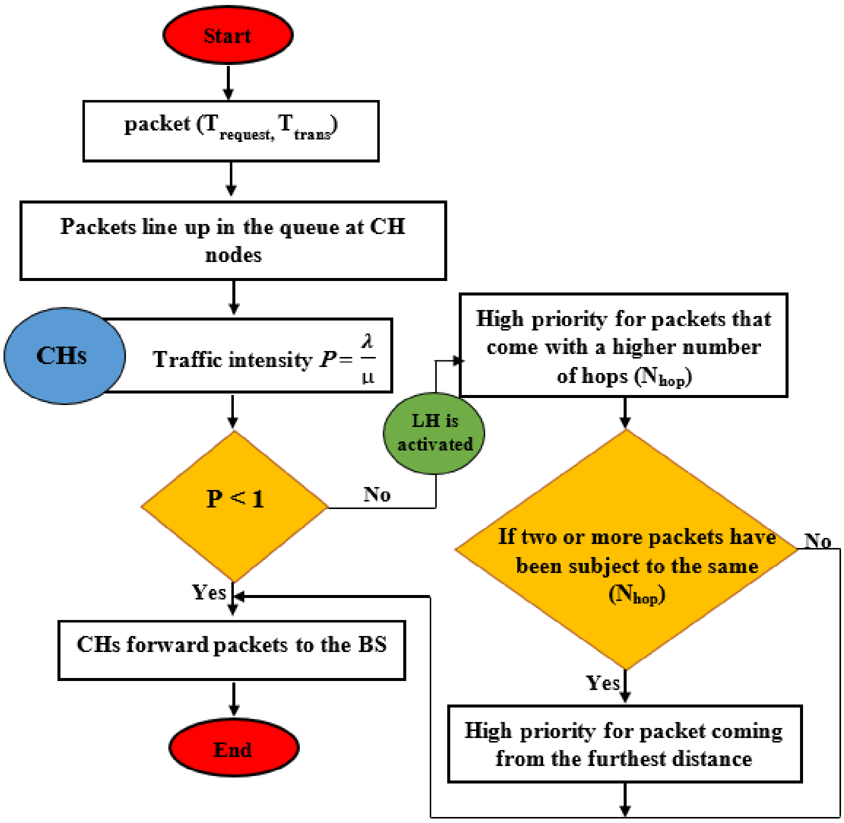


Figure 3.7 Flow-chart of LH procedure.

Algorithm 3.2 : Pseudo-code for LH message scheduling algorithm at CH nodes.

Initialization

Set N_{hops} = number of hops from each node to the BS.

Set d = the distance from each node to the BS.

Set P = traffic intensity

procedure SCHEDULING PROCESS
  For all sensor nodes send data to the BS.
  $\lambda = 1/T_{request}$
  $\mu = 1/T_{trans}$
  Each packet has ($T_{request}$,T_{trans})
  for all $CHs \in network$ **do**
    $P = \lambda/\mu$
    if $P < 1$ **then**
      Forward all packets to the BS
    else
      Sort the packets in terms of hops and distances in descending order
      Forward packets with higher N_{hops} to the BS
      if Two or more packets have $N_{hops} = N_{hops}$ **then**
        Select the packet comes from longer d, as the first forwarded packets to the BS.
      end if
    end if
  end for
end procedure

Figure 3.8 and 3.9 show an example of the LH algorithm and how it works at the CH nodes. We assume that sensor nodes are randomly distributed in the sensing field and they send their packets to the respective CH node through other intermediate nodes. Based on node locations, each packet accesses a different number of hops and distance to reach the assigned CH node. When the packet arrival rate is greater than the CH node service rate (P > 1), the length of the queue increases, the CH node becomes overloaded and drops any extra packets from the queue (i.e., pink and orange packets). In such a case, the LH algorithm rearranges the packets according to the packets with the higher number of hops and longer distance to be forwarded first to the BS (i.e., the blue packet). However, there are two packets in the queue having the same number of hops (i.e., yellow and purple packets), in this case, the proposed algorithm takes the packet that has travelled the longer distance (i.e., the purple packet) to be served first to the BS.

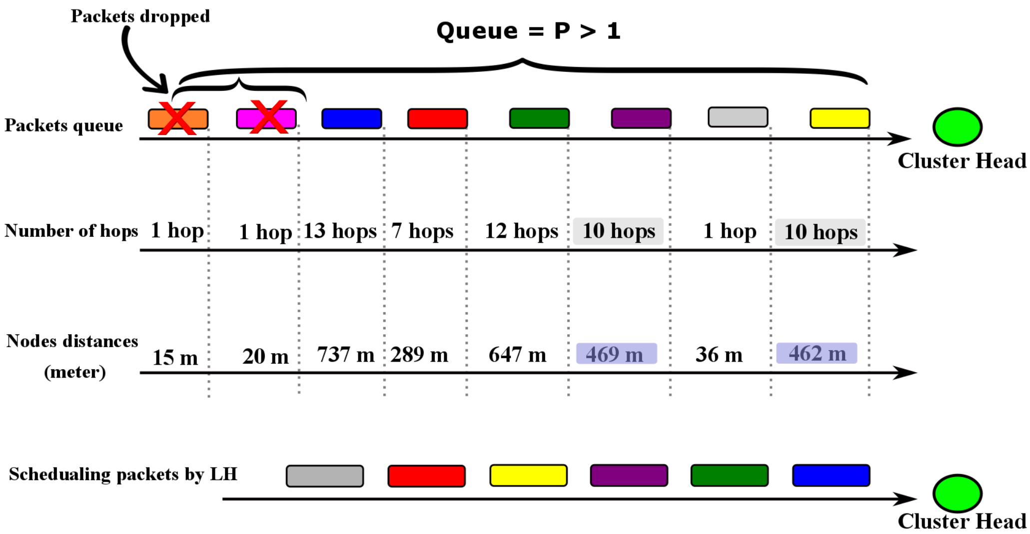


Figure 3.8 An example of LH scheduling algorithm.

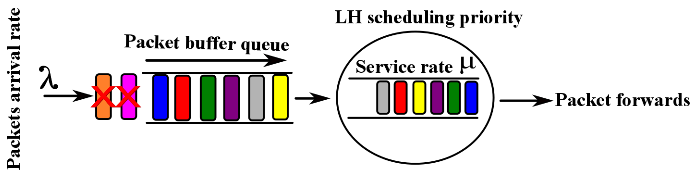


Figure 3.9 Data processing model for the LH scheduling algorithm priority.

3.3.3 Network Buffer Sizing

Sensor nodes have a very low storage capacity which often limits storing to only a few kilobytes [145]. These nodes use this memory during receiving, processing, and transmitting to the target. All of these nodes send their data to their respective CH node which can become overloaded as the CH node memory is not enough to handle all of the packets received. Thus, the probability of packets drop increases due to excessive incoming traffic to the CH nodes.

The rule-of-thumb and Standford rule are the most important rules for dimensioning network queues [146, 147]. Rule-of-thumb states that each link requires a buffer of size $B = RTT \times C$, where C is the bottleneck capacity and RTT is the average round-trip time of the flow passing across the link. This rule is often applied at the edge or cluster devices of the network when the bandwidth capacity and a number of flows are small.

The Stanford rule is used for a large number of TCP flows and higher speed links. The recommended router requires a buffer of size $(RTT \times C)/\sqrt{n}$, where n is the number of TCP flows sharing the bottleneck link [146]. Here the flows at each CH is relatively small, so the rule-of-thumb is assumed for this study.

3.4 Energy Consumption Model

In WSNs, most of the energy is dissipated in the process of data transmission [148]. The total energy consumption of a network consists of the energy consumed by non-CH, intermediate and CH nodes for data aggregation and transmission. Figure 3.10 shows the radio energy dissipation model for WSNs [149, 150]. It is a simple model for radio hardware energy dissipation, where the transceiver dissipates energy to run the radio electronics and the power amplifier.

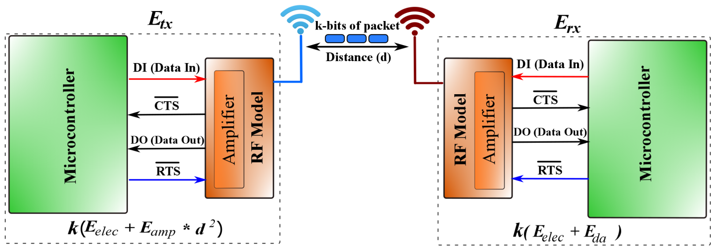


Figure 3.10 The wireless communication power model.

The energy consumed in the model is given as follows [142, 54]:

- To transmit k bits:

$$E_{Tx}(k,d) = k(E_{elec} + \varepsilon_{amp} * d^2) \tag{3.7}$$

- To receive k bits:

$$E_{Rx}(k) = k(E_{da} + E_{elec}) \tag{3.8}$$

Where $E_{Tx}(k,d)$ is the energy used to transmit k bits from the source node to the next-hop node, where d is the distance between the sender and receiver nodes. $E_{Rx}(k)$ is the energy depleted to receive k bits, E_{da} is the energy used for data aggregation and compression. E_{elec} is the energy needed to run the receiver and transmitter circuitry, which depends on factors such as modulation, digital coding, filtering, and spreading of the signal.

ε_{amp} is the energy consumption of the power amplifier per bit, and can be calculated using Equation 3.9:

$$\varepsilon_{amp} = \begin{cases} \varepsilon_{fs} * d^2 & d \le d_0 \\ \varepsilon_{mp} * d^4 & d > d_0 \end{cases} \tag{3.9}$$

Where ε_{fs} and ε_{mp} are the free space propagation (d^2 power loss) and the multi-path fading (d^4 power loss) channel models, respectively. These values depend on the distance between transmitter and receiver, d_0 is a threshold value which can be calculated by using Equation 3.10 [151]:

$$d_0 = \sqrt{\frac{\varepsilon f_s}{\varepsilon_{mp}}} \tag{3.10}$$

Energy consumption of the sensor nodes in the sensing field is classified into three types [54]:

(i) **Non-CH nodes:** gather, k-bits of data from the environment and transmit them to a next-hop node or direct to the CH node. The energy dispatched by a non-CH node (E_{non-CH}) can be calculated as:

$$E_{non-CH} = E_{Tx}(k, d_i) + E_{GPS} \tag{3.11}$$

Where E_{GPS} and d_i are the power dissipations in the global positioning system and distance between non-CH node and its CH node, respectively.

(ii) **CH nodes:** collect and compress the data that comes from the non-CH nodes, and then forwards them to the BS. Hence, the total energy consumed by each CH node can be calculated using Equation 3.12, where M is the total number of sensor nodes that send their packets to the given CH node and d_i is the distance between the CH node and the BS:

$$E_{CH} = M(E_{Rx}(k) + E_{Tx}(k, d_i)) + E_{GPS} \tag{3.12}$$

(iii) **A hop node (intermediate node):** consumed its energy while receiving, processing and transmitting of packets from other nodes. A hop node transmits and receives the information for L number of non-CH nodes. The energy consumption by a hop node E_{hop} can be calculated as:

$$E_{hop} = L(E_{Rx}(k) + E_{Tx}(k, d_{hop,CH})) + E_{GPS} \tag{3.13}$$

We can conclude that energy consumption for transmission is proportional to the square of the distance. Therefore, the aim of the proposed scheme is to reduce the transmission distance between nodes in the sensing field and thus extend the network lifetime.

3.5 Computational Complexity Analysis

The complexity of the proposed algorithms can be analysed in terms of storage and computational complexity. Sensor nodes are restricted to the memory capacity of the processing unit [59]. It is important to reduce the burden on this processor unit to prevent the errors caused by data processing. The time complexity is measured by counting the number of operations performed in the proposed algorithms. We use the Big-O notation to classify the proposed schemes based on their running time [152]. The time complexity of the SPFL routing protocol is $(3n^2+n)$, where n is the number of active nodes in the sensing field. While the time complexity of the LH algorithm is (n^2+8n). The combination of both complexities is $(4n^2+9n)$. An algorithm will be efficient when this function value is small. Therefore, the time complexity is obtained as $O(n^2)$, which is similar or better than other protocols which have complexity $O(n^2)$ and $O(n^3)$, respectively.

3.6 Results and Discussion

This section presents the results obtained from the proposed routing and message scheduling algorithms, and compared to those from state-of-the-art algorithms. Note, all the simulations were repeated 20 times using Matlab for different network topologies with the sensor nodes distributed randomly at the start of each simulation. The averages of these simulations were computed and the obtained results are shown in each section below.

3.6.1 Energy Oriented Path Algorithm

In this section, the proposed scheme (SPFL) is compared to the Energy-Aware Multi-hop Routing (EAMR) [86] and Cluster Heads-Enhanced Hybrid, Energy-Efficient Distributed (E-HEED) protocol [92].

The EAMR scheme is one of the LEACH protocols which divides the sensing field into a number of fixed sub-clusters. Each cluster has a number of sensor nodes and a single CH node. The mechanism of selection of the CH nodes is similar to the LEACH protocol, however, a new CH node is only implemented when the current CH node is dead. Multi-hop routing is used as communication between the nodes.

The E-HEED is one of the HEED modification protocols. In this protocol, the selection of the CH node is based on the HEED protocol. The CH nodes located far from the BS forward their data to another CH node to reach the BS. More details about the proposed EAMR and E-HEED protocols are given in Section 2.4.2.1.

- **The proposed scheme VS EAMR**

  The comparison is based on the parameters shown in Table 3.1 [86].

Table 3.1 Parameters used in the simulation.

Parameter	Value
Electronics energy (E_{elec})	50 nJ/bit
Initial energy of node (E_{init})	2 J
Energy for GPS receiver (E_{GPS})	20 nJ/bit/signal
Energy for data aggregation (E_{da})	5 nJ/bit/signal
Communication energy (ε_{mp})	0.0013 pJ/bit/m^4
Communication energy (ε_{fs})	10 pJ/bit/m^2
Threshold value of distance (d_0)	87 m
Packet size	6400 bits
Packet size for control info	200 bits
Threshold value	0.05 J
Number of nodes (N)	100, 200
Sensing field area	100 m^2

To start with, Figure 3.11 shows the total residual energy in the network after each round when the sensing field had an area of 100 m^2 with 100 nodes. The energy was calculated for each node by including the energy used for collecting, transmitting and receiving packets. However, the energy required for the initial gathering of sensor and node data to enable setting up the protocol to work has not been taken into the calculations of the energy consumed. It appears that EAMR and the proposed scheme consume much the same amount energy till round number 2500. However, from this round, the EAMR outperforms the proposed scheme. This is because all nodes in the proposed scheme send their data to the BS via one or two hops to the BS. This increases the transmission distances which in turn reduces the network lifetime.

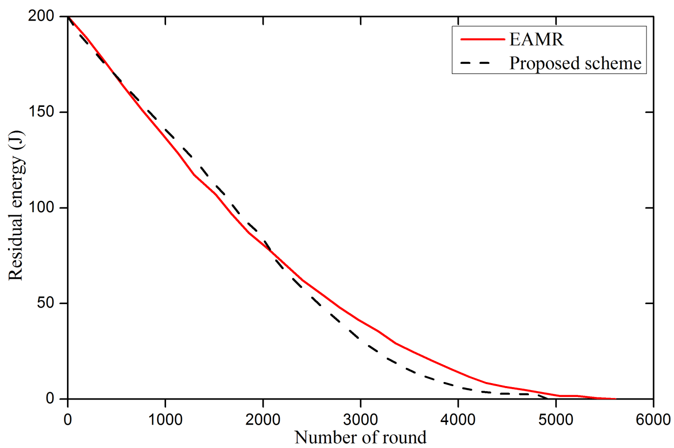


Figure 3.11 Total residual network energy (100 m^2 and 100 nodes).

Figure 3.12 shows the number of nodes remaining alive after a given number of rounds for EAMR and the proposed scheme. It is seen from the figure that the

EAMR achieves a longer network lifetime with 100 nodes and 100 m^2 sensing area compared to the proposed scheme.

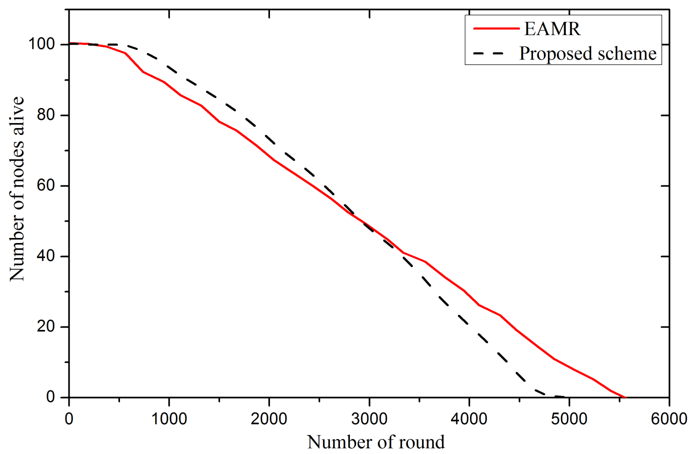


Figure 3.12 Nodes remaining active after each round (100 m^2 and 100 nodes).

Figure 3.13 and 3.14 illustrate the simulation results for 200 nodes and same sensing area 100 m^2.

When the number of nodes within the same sensing area increases, it is expected that the mean distance between nodes decreases. As a result, this reduced energy consumption and extended the lifetime of the nodes. Each node has a different number of forwarding nodes. The proposed protocol balances the load traffic between the non-overloaded and overloaded nodes on each path. This helps to reduce energy consumption and prolong the network's lifetime. We see the proposed scheme has a longer network lifetime compared to the EAMR (see Figure 3.13).

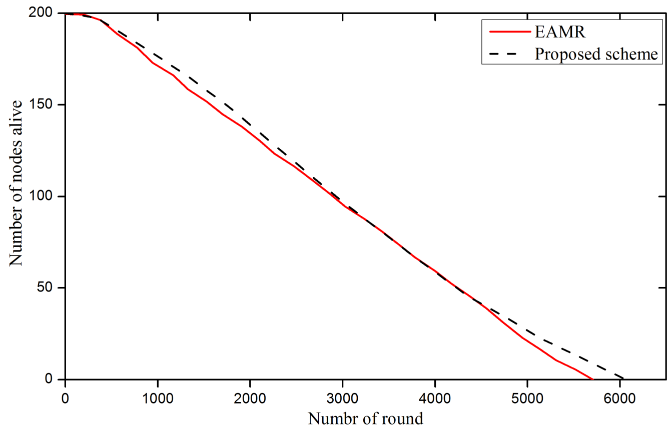


Figure 3.13 Nodes remaining active after each round (100 m^2 and 200 nodes).

Figure 3.14 shows in which round the fist, half and last nodes die. It appears that the first and the half nodes die at the same round for both protocols. However, the last node of the proposed scheme expires after last node of the EAMR. Therefore, the proposed scheme has longer network lifetime as the last node stays active approximately 5 % longer than the EAMR.

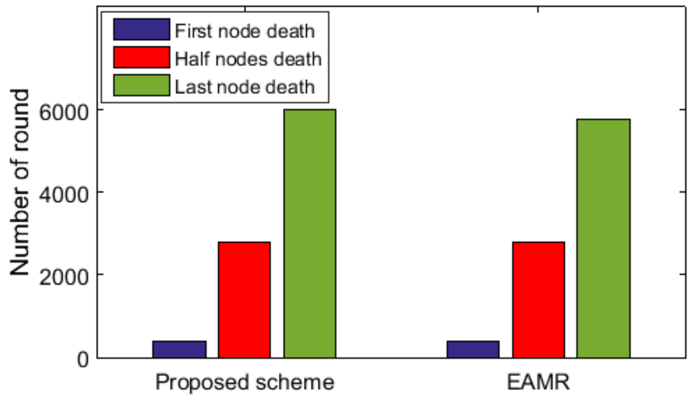


Figure 3.14 First, half and last node death (100 m^2 and 200 nodes).

– **The proposed scheme VS E-HEED**

The comparison is based on the parameters shown in Table 3.2 [92].

Table 3.2 Parameters used in the simulation.

Parameter	Value
Electronics energy (E_{elec})	50 nJ/bit
Initial energy of node (E_{init})	0.5 J
Energy for GPS receiver (E_{GPS})	20 nJ/bit/signal
Energy for data aggregation (E_{da})	5 nJ/bit/signal
Communication energy (ε_{mp})	0.0013 pJ/bit/m^4
Communication energy (ε_{fs})	10 pJ/bit/m^2
Threshold value of distance (d_0)	87 m
Packet size	3000 bits
Number of nodes (N)	300
Sensing field area	300 m^2

Figure 3.15 demonstrates the total energy remaining in the network after each round. The energy was calculated for each node by including the energy used for collecting, transmitting and receiving packets. However, the energy required for the initial gathering of sensor and node data to enable setting up the protocol to work has not been taken into the calculations of the energy consumed. The figure shows that the proposed scheme achieves better energy saving than the E-HEED by 30 %. This is because the proposed protocol takes a decision to balance the traffic load, shifting the traffic away from the node that has higher number of forwarding nodes, and thereby reduce the energy consumption in the overall network.

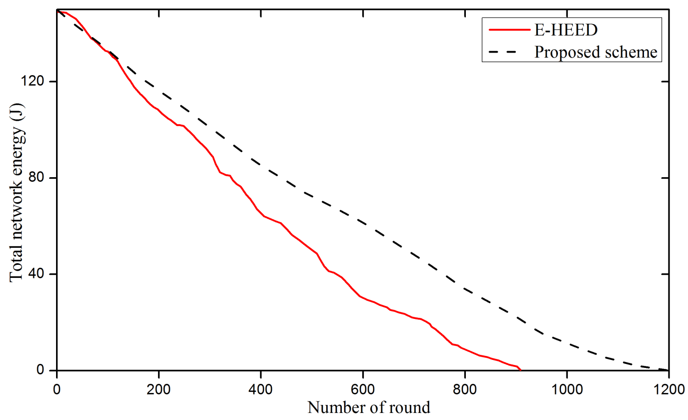


Figure 3.15 Total residual network energy (100 m^2 and 100 nodes).

Figure 3.16 presents the number of alive nodes remaining after a given number of rounds for each protocol. It is clear that, for both protocols, all nodes remain alive until round 400. However, from round number 400 to almost 800, the E-HEED has greater number of alive nodes. From round 800, the proposed scheme has more live nodes than E-HEED. This is because the proposed scheme preserves network energy by balancing the load traffic on each path, transferring from overloaded nodes to others. Thus, the proposed scheme achieves a longer network life than the E-HEED.

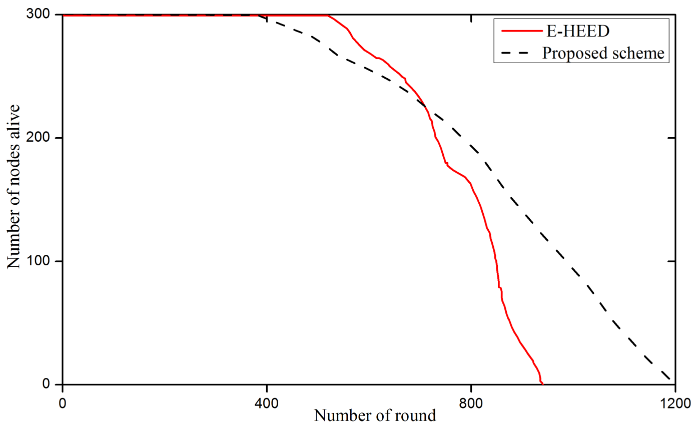


Figure 3.16 Nodes remaining active after each round (100 m^2 and 200 nodes).

Figure 3.17 shows the number of rounds at which the first, half and last nodes die. It appears that the rounds for the proposed scheme are higher than that for the E-HEED protocol. According to the Figure 3.17, the proposed scheme improves the network lifetime over E-HEED by 30 %. However, the first node of the proposed algorithm dies before the first node of the E-HEED, while the half nodes of the proposed algorithm stay alive longer than that of the E-HEED.

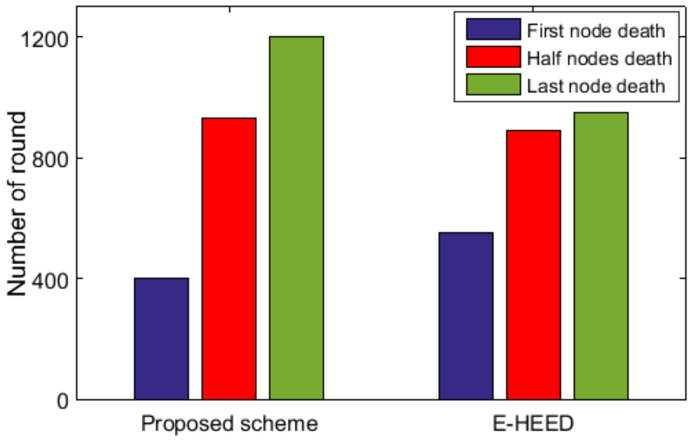


Figure 3.17 First, half and last node death (100 m^2 and 200 nodes).

3.6.2 Message Scheduling Approach

This section presents the results obtained from the LH scheduling algorithm, benchmarked against algorithms widely used today: Earliest Deadline First (EDF) [124], Tree-Based Mobile Sink (TBMS) [54], Multi-core Processor Technique [126].

The EDF is a scheduling mechanism which is used to manage the real-time tasks in the queues for WSNs. This method assigns high priority for packets which are nearly expired. On the other hand, the TBMS protocol creates a tree-cluster routing structure with a mobile sink to collect the data from the sensor nodes in the sensing field. The TBMS aims to minimise the transmission distances between sensor nodes and thereby extend network lifetime. However, this protocol ignores the retransmission of packets caused by the limited resources of the CH nodes.

The third method is the multi-core processor technique. This method carries out multiple tasks at the same time and thus prevents the retransmission of packets. In this study, the sensor nodes in the LH, EDF and multi-core protocols followed the SPFL routing algorithm to forward their data to the BS. Further details about the EDF, TBMS and multi-core processor technique are given in Section 2.4.2.1.

All parameters which are used in the simulation of this subsection are listed in Table 3.3 [54, 143]. Additionally, we considered four different scenarios for the validation of the effectiveness of the proposed scheme, see Table 3.4.

Table 3.3 Parameters used in the simulation

Parameter	Value
Electronics energy (E_{elec})	50 nJ/bit
Initial energy of node (E_{init})	0.25 J
Energy for GPS receiver (E_{GPS})	20 nJ/bit/signal
Energy for data aggregation (E_{da})	5 nJ/bit/signal
Communication energy (ε_{mp})	0.0013 pJ/bit/m^4
Communication energy (ε_{fs})	10 pJ/bit/m^2
Threshold value of distance (d_0)	87 m
Arrival rate per sensor node	0 to 0.1 sec.
Service rate	0.5 sec.
Buffer size	202 bytes
Packet size	2000 bits
Retransmission overhead size	200 bits
Number of nodes (N)	100, 300, 500
Sensing field area	200 , 500, and 1000 m^2

Table 3.4 Different scenarios used in the simulation

Scenarios	No. of nodes	Sensing area
1	100	200 m^2
2	100	500 m^2
3	300	500 m^2
4	500	1000 m^2

Figure 3.18 shows the total residual energy after each round when the sensing area is 200 m^2 with 100 nodes. The total residual energy is the sum of the remaining energy at all sensor nodes after each round in the network. In this figure, when the number of rounds was less than 300, it is clear that all methods have the same total residual energy. However, for a greater number of rounds, the TBMS has slightly higher residual energy than the other two techniques. This is because the mobile sink covers all the sensing area and thus, reduces the transmission distances between nodes, reducing the transmission power of the sensor nodes.

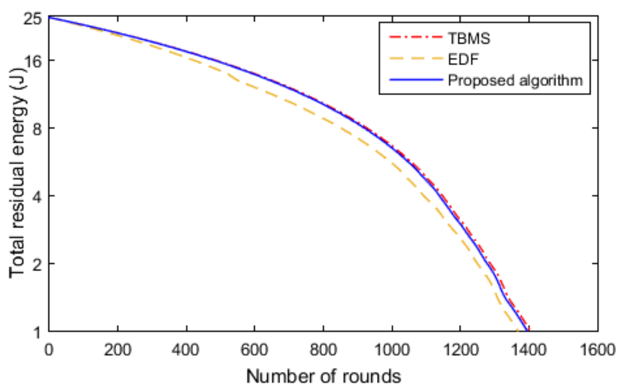


Figure 3.18 Total residual energy (200 m^2 sensing field and 100 nodes).

The sensing area and number of nodes were extended as in Table 3.4 in order to prove that the proposed scheme is better suited for large-scale networks. It is obvious from Figures 3.19 to 3.21 that the proposed method achieves better energy saving than EDF and TBMS algorithm, by as much as 1 % and 2 %, respectively, when the number of nodes is 100 and the sensing area is 500 m^2. However, for higher number of nodes and larger sensing area (i.e., 500 nodes and 1000 m^2), it achieves 8 % and 29 % energy saving compared to EDF and TBMS, respectively. This is because the SPFL protocol balances the load traffic between the nodes on the path. Furthermore, the LH algorithm gives higher priority for packets that consume more energy so there is no need for retransmission. As a result, the proposed algorithm extends the network lifetime. The other observation is that the multi-core technique is superior to the other methods. However, the proposed protocol has almost the same energy performance when the number of rounds is relatively high.

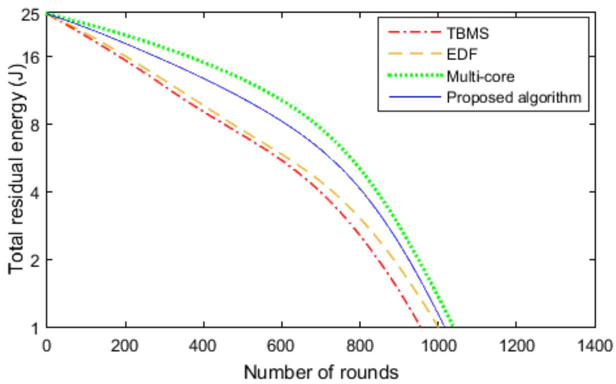


Figure 3.19 Total residual energy (500 m^2 sensing field and 100 nodes).

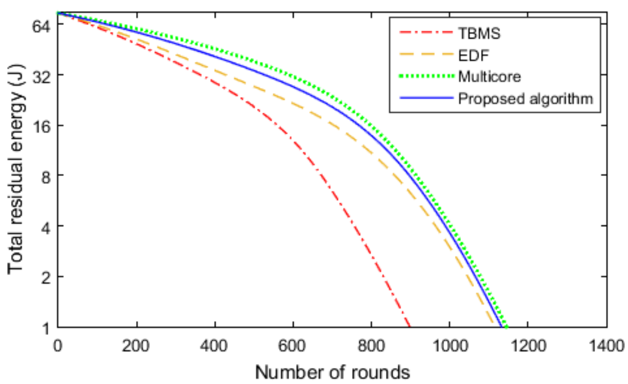


Figure 3.20 Total residual energy (500 m^2 sensing field and 300 nodes).

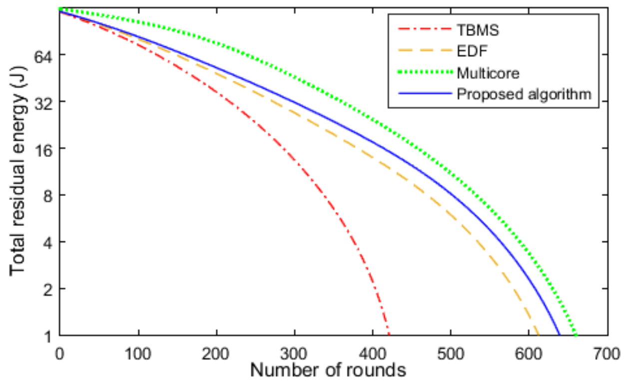


Figure 3.21 Total residual energy (1000 m^2 sensing field and 500 nodes).

Figure 3.22 shows the average transmission distances (including the distances for retransmission packets) for the four algorithms based on the scenarios presented in Table 3.4; where the average transmission distance is the total transmission distances form all nodes to the BS divided by the number of sensor nodes. As previously mentioned, the protocols with lower transmission distance are more energy efficient than the ones with greater transmission distance. It is clear from the figure that the average transmission distance of the proposed scheme is less than for the EDF and TBMS for all of the considered scenarios. However, it is also noticeable for all scenarios, the SPFL-LH and multi-core protocols have almost the same average transmission distances.

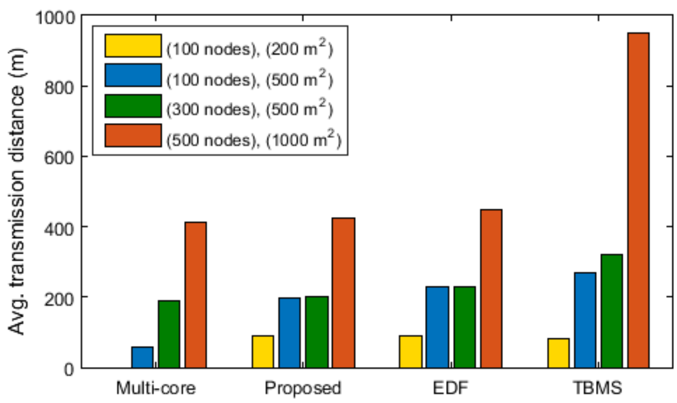


Figure 3.22 Average transmission distance (m).

Figure 3.23 shows the average number of hops (including the hops of retransmission packets) for each of the four schemes based on scenarios shown in Table 3.4. The average number of hops is the summation of the number of hops between the sensor nodes and the BS divided by the total number of nodes in the sensing field. For the first scenario, it is clear that the average number of hops in the TBMS protocol is less than that in the proposed and EDF algorithms. This means the TBMS is more energy efficient. However, for the other three scenarios, the proposed algorithm is superior to the EDF and TBMS methods. It is also obvious the multi-core and the proposed scheme have almost the same average number of hops for all of the scenarios considered.

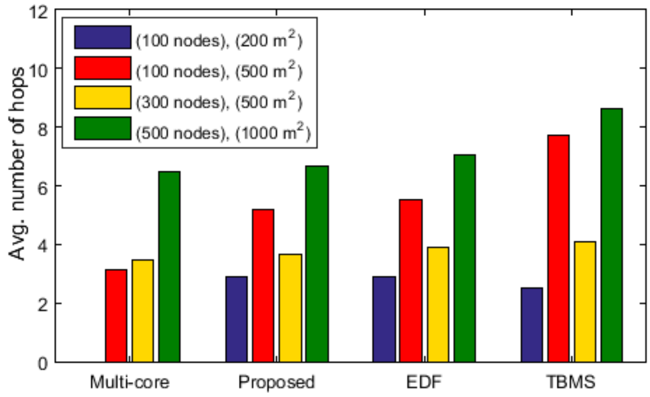


Figure 3.23 Average number of hops.

Figure 3.24 shows the average energy consumption for the first transmission for the four schemes based on the scenarios of Table 3.4. It is obvious that the average energy consumption for all four algorithms increased when the number of nodes and sensing area increased. However, in accordance with second, third and four scenarios of Figures 3.22 and 3.23, the proposed scheme has lower transmission distance and smaller number of hops than EDF and TBMS which leads to lower energy consumption. For example, for the fourth scenario, the proposed scheme achieves 26% and 50% energy saving compared to the EDF and TBMS algorithms, respectively.

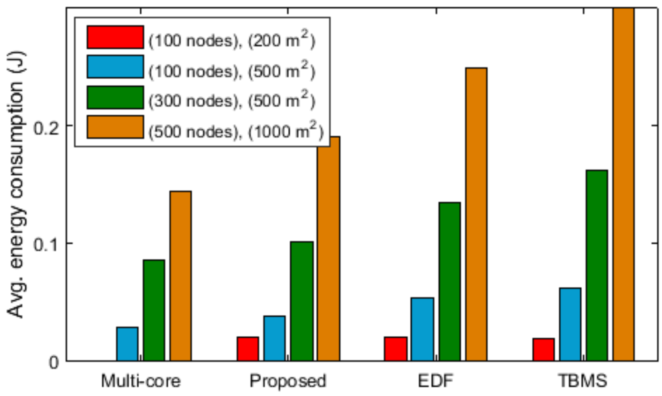


Figure 3.24 Average energy consumption for first transmission (J).

The last sets of results shows a comparison between the proposed scheme and the other protocols in terms of the first node death (FND), half nodes death (HND), last node death (LND), transmission time, number of transmitting and receiving packets and throughput based on Table 3.4 scenario 4.

Figure 3.25 shows the rounds at which FND, HND and LND occur for the four schemes. When a node dies due to energy depletion, it becomes a non-connected node. Thus, some sensor nodes transfer their data through a longer distance and hence, increase energy consumption. From these figures, it is obvious that the FND, HND, and LND of the proposed scheme occur after that in EDF and TBMS methods which means the proposed scheme is more energy efficient and thus prolongs the network lifetime. However, the HND and LND for the multi-core and the proposed algorithm happen almost the same round.

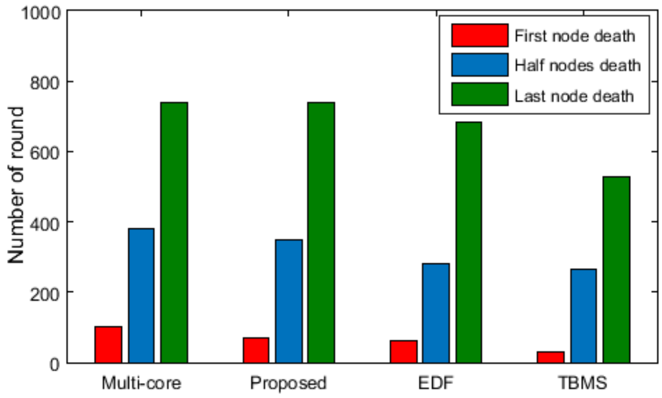


Figure 3.25 First, half and last node death for each of the four algorithms (TBMS, EDF, Multi-core and the one being proposed here).

Figure 3.26 shows the average transmission time for the all the algorithms considered. The transmission time is the time taken by the packet to travel through the communication medium from the source node to the ultimate receiver. During the multi-hop transmission process, each sensor requires the processing time to send and receive data. The simulation setting adopted was based on [54, 154], which took 2 ms for a sensor node to make a transmission. All sensor nodes updated their information each interval period. The length of an interval for update packets is 200 ms. The propagation delay is calculated by: ($transmission\ distance\ /\ speed\ of\ light$). Therefore, the total transmission time is: ($propagation\ delay + processing\ time\ for\ each\ transmission$). It appears from this figure that the average transmission time of the proposed scheme is less than that of the TBMS and EDF. This is because the proposed scheme has lower transmission distances compared to these two protocols.

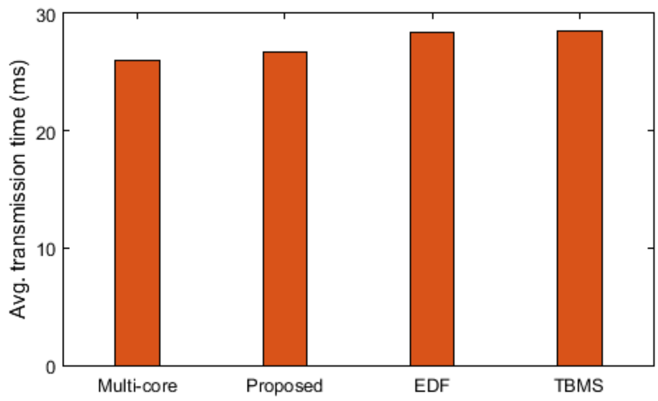


Figure 3.26 Average transmission time (s) for each of the four algorithms (TBMS, EDF, Multi-core and the one being proposed here).

Figure 3.27 presents the average number of receiving and transmitting packets of each protocol. As previously mentioned, the intermediate nodes consume most of their energy during the receiving, processing and transmitting process. Thus, reducing the number of hops minimises the number of transmitting and receiving packets in each node on the path. This minimises the total energy consumption of the network and thereby increases its life. The proposed protocol has a lower average number of received and transmitted packets compared to that of EDF and TBMS, but slightly more than the multi-core.

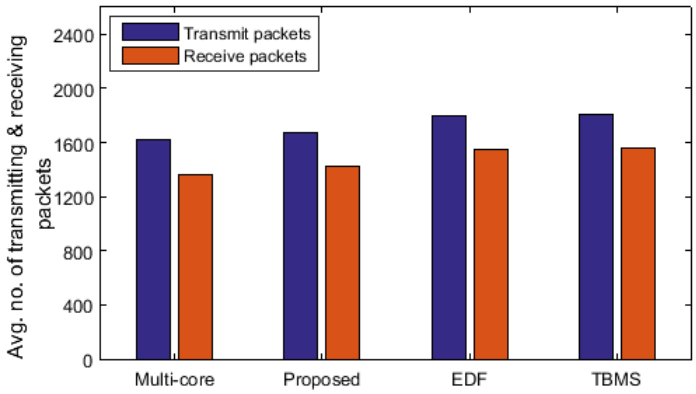


Figure 3.27 Average transmitting and receiving packets for each of the four algorithms (TBMS, EDF, Multi-core and the one being proposed here).

Figure 3.28 reveals the throughput of the proposed algorithm compared to the three alternatives, where the throughput is the percentage of successfully delivered packets sent from the sensor nodes to the BS each round. As can be seen from this figure that the TBMS has the worst throughput, while the multi-core technique has the best. The multi-core protocol is superior to the other protocols, but requires extra memory to handle the packets during processing. Additionally, overheating due to the use of two processors can cause the damage to the device [127].

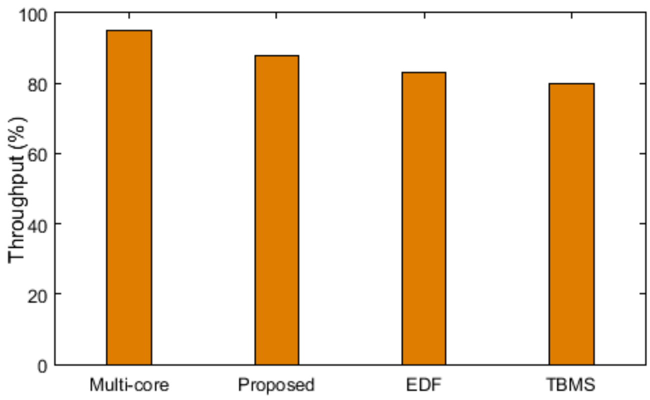


Figure 3.28 Throughput (%) for each of the four algorithms (TBMS, EDF, Multi-core and the one being proposed here).

3.7 Summary and Contributions to Knowledge

Energy-saving is one of the most essential requirements for WSNs and IoT networks, as batteries are usually the main source of power for these networks. This chapter has presented a new routing protocol and a novel scheduling algorithm to minimise energy consumption and prolong network life. The SPFL protocol balances the network load between the nodes on the forward path by intelligently selecting the next-hop node with least transmission distance and a smallest number of forwarding nodes to the BS. While the LH scheduling algorithm assigned high priority to messages coming from nodes that are located farther away and have accessed a higher number of hops, to be served first at CH nodes in order to prevent these messages from being retransmitted.

The performance of the SPFL protocol was evaluated in terms of power consumption and network lifetime. The proposed protocol achieved improvements of 30 % and 5 % in energy conservation compared to E-HEED and EAMR protocols, respectively. When tested under the conditions specified in Table 3.4, scenario 4, the SPFL-LH algorithm achieved 29 % and 8 % improvements in energy conservation compared to TBMS and EDF algorithms, and also reduced the average energy consumption each round by 50 % and 26 % compared to the TBMS and EDF protocols.

A comparisons between the SPFL-LH algorithm and the multi-core technique were also provided in this chapter. It was shown that the performance of the multi-core protocol is slightly better than the proposed algorithm for all the considered scenarios. This is because there is no retransmissions of packets in this protocol. However, the problem with the multi-core processor system is that it requires a large memory to hold the packets during processing. Additionally, overheating due to the use of two processors can cause device damage.

The major contributions of this chapter are summarised as:

- Multi-hop communication reduces the transmission distances between the nodes and thus extends the network lifetime.
- An energy oriented path selection strategy is proposed which selects the next-hop node as the one with least transmission distance and fewer forwarding nodes in order to balance network traffic and prolong network lifetime.
- A simulation study of location of sensor nodes and number of hops, and their relationship with energy consumption was carried out. It revealed that packets sent from nodes located furthest from the BS and accessing a higher number of hops consume more energy during the transmitting and receiving process.

- A new scheduling algorithm is introduced which assigns high priority for packets that consume more energy to be served first at the CH nodes and this prevents these packets from being retransmitted.

In the next chapter, a new routing strategy is introduced that take into consideration the number of neighbour nodes around each node on the forward path when selecting the next-hop node. This will be shown to help to reduce energy consumption and maximise network lifetime.

Chapter 4

Energy and Geo-location Based Modelling

Increasing the number of neighbour nodes around a node has a negative impact on the network lifetime of WSNs. This is due to the adverse effects caused by the overhearing. This chapter introduces an algorithm which combines a new routing technique and a distributed clustering formation to help overcome this problem. The algorithm also addresses the interference due to overhearing, reduces energy consumption and thus extends the network lifetime.

4.1 Introduction

In dense WSN deployment, each node can have a different number of neighbour nodes located within its transmission range (TR), the maximum distance that a node can communicate its data to other nodes. A transmission from a source node is potentially overheard by all other neighbour nodes located within its TR, even if these nodes are not the intended destination [66, 67]. Of course, neighbour nodes can be at different transmission distances from each other in the sensing field.

A source node also receives data from surrounding nodes when they transmit their data. However, a node with more neighbour nodes, especially when these neighbours are located at longer distances, and require a greater value of TR will consumes more energy than a source node with smaller neighbour nodes and within a shorter distance. This is because increasing the transmission distances between a node and its neighbour nodes will consume more energy, so the power required for transmission increases rapidly with distance (see Figure 4.1). *Nodes A* and *B* have a different number of neighbour nodes located within different transmission distances. However, *Node A* has a higher number of neighbour nodes that are located at further distances. This increases the required transmission power of *Node A*, which leads to more energy consumption at this node compared to *Node B*.

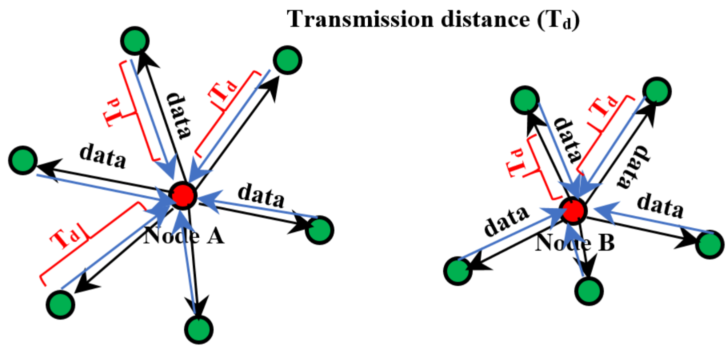


Figure 4.1 Neighbour nodes and transmission distance.

These nodes usually use the unlicensed frequency band 2.4 GHz to communicate with each other [155]. This particular band will often suffer from interference generated by other nearby networks working in the same frequency range, with a level of interference that increases with increase in the number of neighbour nodes [70]. Because the 2.4 GHz frequency band is overcrowded and limited in capacity this will increase the probability of packet collision which leads to more packet losses and thus increased number of retransmissions [156]. Retransmission increases the complexity and number of overhead transmissions for the network, and thus increases energy consumption [157]. Therefore, a more suitable routing strategy needs to be adopted for disseminating the data via a suitable node/path which balances the load within the network by considering the number and locations of the neighbour nodes around each node [158].

The selection of cluster head (CH) nodes, and their locations, are very important for node management, to minimise energy consumption, improve load balancing and thereby lengthen the network lifetime [56]. The choice of the CH nodes should, ideally, be based on the shortest transmission distances between the node and the BS which reduces energy consumption.

In the previous chapter, the routing technique differentiated between the nodes on the path, and balanced the network load by selecting the next-hop node with fewest forwarding nodes. However, in this chapter, a more advanced energy and geo-location based modelling of route selection is proposed. This scheme considers the transmission distances from a node to all neighbour nodes that are within its TR. The interference measurement approach is adopted to select the next-hop node, and CH node selection is based on the transmission distances to the BS. The nearest node to the BS in a sub-cluster is elected as CH node for that sub-cluster. By doing this, the proposed scheme reduces the transmission distances from a CH node to the BS and thus helps maximise the network lifetime. The proposed scheme finds an efficient path by which to forward the data to the BS which, in turn, helps minimise the energy consumption of the entire network.

4.2 Impact of Interference and Packet Error Rate on the Energy Consumption

In the following subsections, we first give and evaluate the relation between interference and energy consumption, as it affects the lifetime of the network. Then, we discuss the effect of packet error rate on the WSNs communication.

4.2.1 Interference Estimation

In WSNs, the TR has a directly proportional relationship to energy consumption. All neighbour nodes, within TR of a source node, apart from the destination node see the transmission of the source node as interference [159]. It is possible that the wireless transmission from one device can affect the reception of surrounding devices that are located within its transmission distance [160]. The overall effect of the interference depends on the number of neighbour nodes within the TR of the source node, as well as the locations of those nodes around the source node. A node with a larger number of neighbour nodes can overhear the transmissions from a higher number of nodes, and this causes greater interference [67]. Also, the greater transmission distance, not only will more transmission power be required [161] but more nodes will become neighbour nodes which raises the possibility of increasing the interference effect. This will result in degrading the communication quality, increasing data transmission failure and decreasing the network lifetime.

To correctly estimate interference $I_{(i)}$ which is generated in a node and caused by its neighbour nodes, the power from each neighbour node received at this node should be taken into consideration, see Equation 4.1 [162].

$$I_{(i)} = \frac{N(p)}{N(P_{max}) + \beta} \sqrt{\frac{P^2 + P_{max}^2}{2P_{max}^2}} \tag{4.1}$$

Where $N(P)$ is the number of neighbour nodes which are reachable within the normal transmit power P. $N(P_{max})$ is the largest neighborhood, for example, all nodes reachable with the maximum power level P_{max}. β is a dimensionless correction factor which is required to distinguish nodes which utilise maximum transmission power but have a different number of neighbour nodes. The optimum value of this factor is ≥ 1.

In multi-hop communication, a source node depends on other intermediate nodes to forward its packets to the destination which is located outside its TR. These intermediate nodes act as relays for packet retransmissions. Each relay node has number of neighbour nodes connected to it. Increasing the number of neighbour nodes in each relay on a path leads to an increase in the collisions between wireless transmission signals in this particular path. Thus, it increases packet drops and increases both energy consumption and interference. The cost in term of estimated interference for a path $I(path)$ can be summarised as:

$$I(path) = \sum_{\forall (i,j) \in path} I_{i,j} \tag{4.2}$$

Where $I_{i,j}$ is the estimated interference that is caused by the transmission from node i to node j in a path.

Based on Equations 4.1 and 4.2, the number of neighbour nodes and transmit power levels are the main parameters that affect the interference. A formal assessment of these properties are:

- The lower the power (P) used to transmit a packet from node i to j, the less interference is produced. Assuming that the number of neighbour nodes $N(P)$ is the same for each node involved in the transmission path, the node with less transmission power P produces less interference I_i.
- The node which is surrounded by fewer neighbour nodes experiences less interference than nodes that have more neighbour nodes. Assuming that the transmit power P is the same for each node involved in the transmission path, the node

with higher number of neighbour nodes $N(P)$ experiences greater interference, I_i, than nodes with fewer neighbour nodes.

Figures 4.2 and 4.3 show the simulated interference produced at a node, and energy consumed for different transmission paths. Both figures were generated based on parameters listed in Table 4.1 [54] and Table 4.2 (scenario 4).

Table 4.1 Parameters used in the simulation.

Parameter	Value
Initial energy of node (E_{init})	0.25 J
Electronic energy (E_{elec})	50 nJ/bit
Energy for GPS (E_{GPS})	20 nJ/bit/signal
Energy of data aggregation (E_{da})	5 nJ/bit/signal
Communication energy (ε_{mp})	0.0013 pJ/bit/m^4
Communication energy (ε_{fs})	10 pJ/bit/m^2
Threshold value of distance (d_0)	87 m
Packet size	4000 bits
Advertise message size	200 bits
Number of nodes (N)	100, 300, and 400
Sensing field area	200, 300, and 800 m^2

Table 4.2 Different scenarios used in the simulation.

Scenario	No. of Nodes	Sensing Area
1	100	200 m^2
2	100	300 m^2
3	300	300 m^2
4	400	800 m^2

Figure 4.2 shows the estimated interference, based on Equation 4.1, for an individual node that has a number of surrounding nodes. The figure was generated by randomly picking ten nodes from the sensing field and giving each node a different number of surrounding nodes. From this figure, we see that a node with a higher number of neighbour nodes consumes a larger amount of energy, as explained above, and experiences higher interference levels. On the other hands, a node with fewer neighbour nodes consumes a smaller amount of energy and has less interference. This is clearly seen when comparing the results for 20 and 3 neighbour nodes in Figure 4.2.

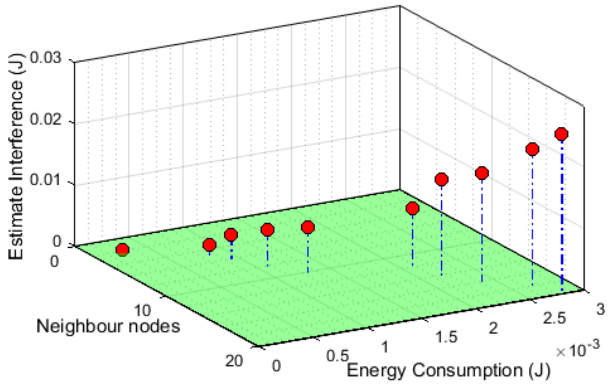


Figure 4.2 Estimated interference vs no. of neighbour nodes and energy consumed.

Based on Equation 4.2, we give an example of the estimated interference for different paths in a sensing field, with each path having a different number of hops and neighbour nodes. We see from Figure 4.3 that, generally, a path with a higher number of neighbour nodes has greater interference than a path with fewer neighbour nodes. For example, the estimated interference value for the path with 2 hops and 123 neighbour nodes (0.0595 joule), is higher than for the path with 3 hops and 113 neighbour nodes (0.049 joule). This is because in the former case, the node is surrounded by a greater number of neighbour nodes than the latter. We conclude that to reduce interference, it is crucial to use an optimum routing technique that takes into account the minimum number of neighbour nodes during the next-hop node selection process.

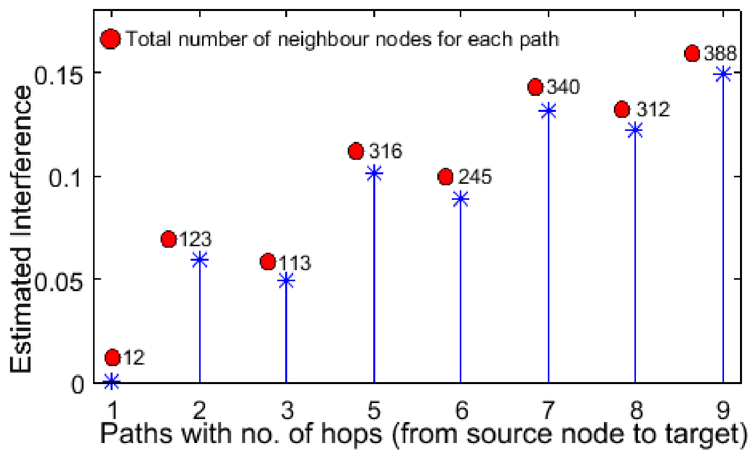


Figure 4.3 Estimated interference for different paths (hops) and number of neighbour nodes encountered.

4.2.2 Packet Error Rate (PER)

In multi-hop communication, the PER increases with the increase in the distance between the transmitter and receiver [163], so a a link with a lower PER generates a lower number of retransmission packets and thus consumes less energy [162].

The expected cost of energy $E(c)$ used for the transmission to succeed is given in Equation 4.3 [164], when a number of hops (N) is equal to 1:

$$E(c) = (1 - r_i) * (N * E_h) + r_i * (N * E_h + N * E_h * \frac{1}{(1 - r_i)^2}) \tag{4.3}$$

E_h is the transmit power which is calculated based on the equation in Section 3.4, r_i is the *PER* from source node n_i to the target node n_{i+1}, $i \in$ (1,I) where I is the total number of nodes.

The probability of packet loss increases with increased number of hops and distance, which leads to a decrease in the number of packets that reach their target. Therefore, when the number of hops is $1 \leq N < I - 1$, Equation 4.4 shows the *PER* along the path over each transmission link i, j as:

$$PER(path) = 1 - \prod_{\forall(i,j) \in path} (1 - PER(i, j)) \tag{4.4}$$

Where $PER(i, j)$ is the packet error rate when sending packets from source node i to the target j belong to the same $path(i, j)$. We can conclude that the expected cost of energy $E(c)$ for a path that has N number of hops is:

$$\begin{aligned} E(c) = \sum_{\substack{i=1 \\ j=i+1}}^{I-1} (i - PER(i, j)) * (N * E_h) + PER(i, j) \\ (N * E_h * \frac{1}{(1 - PER(i, j))^2}) \end{aligned} \tag{4.5}$$

A simple example is shown in Figure 4.4 to demonstrate the increase in average expected route cost with the transmission distances, in terms of the number of hops on the path. The plot was generated by randomly picking six paths from parameters shown in Table 4.1 and Table 4.2 (scenario 4). Each path uses a different number of intermediate nodes and distance when forwarding the data from the source to the ultimate receiver. We assumed that the *PER* is fixed in each hop, so that the expected route cost consumes more energy when the transmission distance and PER increase. As a result, we see that shorting the transmission distances between the nodes on the path increased network lifetime. The evolution results are based on Equations 4.5.

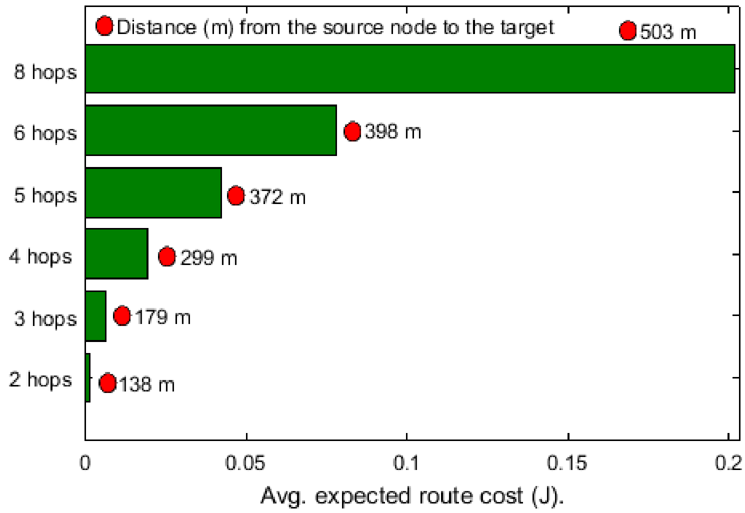


Figure 4.4 Average expected route cost (J) with increased number of hops and distance (m).

4.3 System Model

Here we consider a system composed of a number of sensor nodes which are distributed randomly in the sensing field and clustered into subgroups not necessarily of equal numbers. Each group of nodes has a CH node. All nodes have the same TR and same initial energy level. Each node gathers information from the environment and sends it to their respective CH node. The task of the CH node is to collect, compress, and forward the information to the base station (BS), as in most IoT-enabled WSN application scenarios. The BS acts as the gateway between the sensor nodes and end-users. It is fully powered and placed in the centre of the sensing field as shown in Figure 4.5. The distance (d_i) between two nodes is given by Euclidean geometry (see Equation 3.1). Matlab software was used to simulate the proposed network and Table 4.1 shows the initial parameters of the WSNs for simulating a different number of nodes and sensing areas based on Table 4.2 scenarios [54], On page 90.

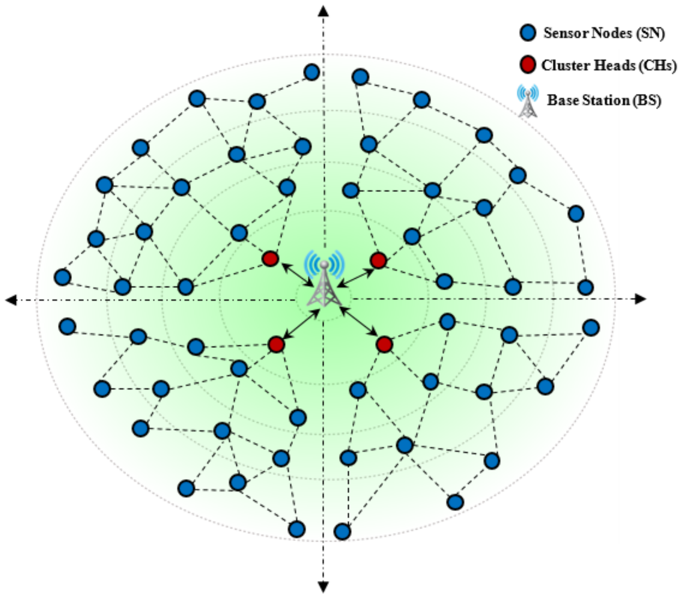


Figure 4.5 Typical WSNs architecture.

4.3.1 Proposed Algorithm

The primary goal of the proposed energy and geo-location algorithm is to reduce transmission distances and balance the power usage among the nodes, and thus increase the network lifetime of WSNs. The proposed scheme is divided into two main phases, but note that neighbour nodes in this algorithm are nodes which are located within the TR of a node and connected to it by a single-hop.

- **Phase 1:** we assume that each node has sufficient initial power to communicate with the others. Sensor nodes are required to exchange a $Hello$ message before the actual data communication begins. The proposed protocol is designed so that sensors establish communication and negotiate the parameters of the network before transmitting data such as sensor location, number of neighbour nodes, and other important details. Thus, a sensor node sends a synchronised packet ($SYNC$) over the network to discover all neighbour nodes. Sensor nodes not located within the TR of a source node will be in sleep mode. The neighbour nodes respond to the $SYNC$ packet and store a sensor node information, they then return a confirmation receipt $synchronise - acknowledgement$ ($SYNC/ACK$), back to the source node. Upon completion of this process, connection has been established between the nodes in the sensing field. The nodes and BS can communicate with each other based on these steps, as shown in Algorithm 4.1, lines 1-12.

  The TR of the nodes and sensing field are based on the parameters shown in Table 4.1, and scenarios shown in Table 4.2.

- **Phase 2:** the three main parameters considered in this phase: transmission distance from the source node to the next hop; transmission distances between each node and their neighbour nodes; and the estimated interference. However, during the path selection, the proposed algorithm gives priority to the path with least transmission distance. In the case where two paths have the same transmission distances, the path selection gives priority to the lower transmission distances between each node and its neighbour nodes. The estimated interference is the last parameter to be considered in the path selection process. The steps of the path selection process are described as follows:

  (1) In multi-hop transmission, nodes use intermediate nodes to forward their packets. The proposed protocol determines the least average hop length to reach the ultimate receiver based on the Equation 4.6:

$$Avg.hop.length_i = \frac{dis(i,j)}{S(i,j)} \tag{4.6}$$

Where $dis(i,j)$ represents the total transmission distances from the source node i to the final destination j and $S(i,j)$ is the total number of nodes on the path, see lines (14-15) in Algorithm 4.1.

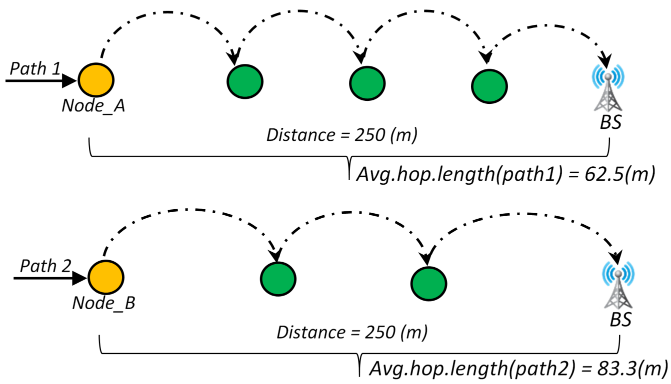


Figure 4.6 Two paths with same overall distance but different hops.

Consider the simplified scenario shown in Figure 4.6. The source nodes *Nodes A* and *B* forward their data to the ultimate receiver via two different paths. The total transmission distance for each path is 250 meters. However, the number of intermediate nodes in these paths are not the same. *Node A* sends its data to the BS via four hops while *Node B* sends its data via three hops.

For $path1$, according to the Equation 4.6, the average hop length is 62.5 m. For $path2$, the average hop length is 83.3 m. Using the equations described in Section 3.4 and parameters given in Table 4.1, we find that the total energy consumed for *Paths* 1 and 2 is 0.00142 and 0.00143 Joule, respectively. *Path* 1 is the best way to transmit the data packets. This is supported by the study [165] which showed that the shorter the average hop length, the less energy consumed and the longer network lifetime.

In case where a source node has two neighbour nodes with the same transmission distances towards the BS, the selection process is more complicated. In this case, a different next-hop selection policy and procedure should be implemented.

We see that the proposed method takes into consideration the transmission distances between a node and its neighbour nodes.

(2) In a such case, the proposed algorithm first calculates the transmission cost (Tc_i) between a single node and its neighbour nodes based on Equation 4.7:

$$Tc_i = \frac{\sum_{i=1}^{N} d_i}{N} \tag{4.7}$$

Where d_i is the sum of transmission distances between the node and its N neighbour nodes.

Secondly, the proposed method chooses the next-hop node with the lowest Tc_i. This selection process is shown in lines 23-24 of Algorithm 4.1.

Because of the high density of sensor nodes in the sensing field, in some cases, different nodes can have same Tc_i. In this case, estimated interference is considered by the algorithm for the next-hop selection.

(3) In this step, the proposed scheme calculates the estimated interference of nodes that have the same transmission distances and Tc_i. The node that has least estimated interference is selected as the next-hop node based on Equation 4.1. This step is presented in Algorithm 4.1 in lines from 26-30.

When a new node joins the network, it starts by sending a message advertising its presence to its neighbour nodes to establish a connection with them. As soon as, it establishes the connection, it will be considered by the proposed algorithm for minimising transmission distance from the source node to the next hop, between each node and their neighbour nodes, and the estimated interference.

A brief example of the proposed algorithm incorporating the above steps is shown in Figure 4.7. Sensor nodes are distributed randomly in the sensing field. *Node A* forwards its data to the BS through intermediate nodes. *Node B* and *C* are the two nodes to be selected as the next-hop node for *Node A*. However, these two nodes are located with the same transmission distance from *Node A* and they have the same Tc_i. However, the estimated interference of *Node C* is less than that of *Node B*. This is because the former has fewer neighbour nodes than the latter. Therefore, *Node C* is selected as the next-hop node to receive the data from *Node A*. Then, *Node C* selects the next-hop node based on the above criteria and so on until the data reaches the BS.

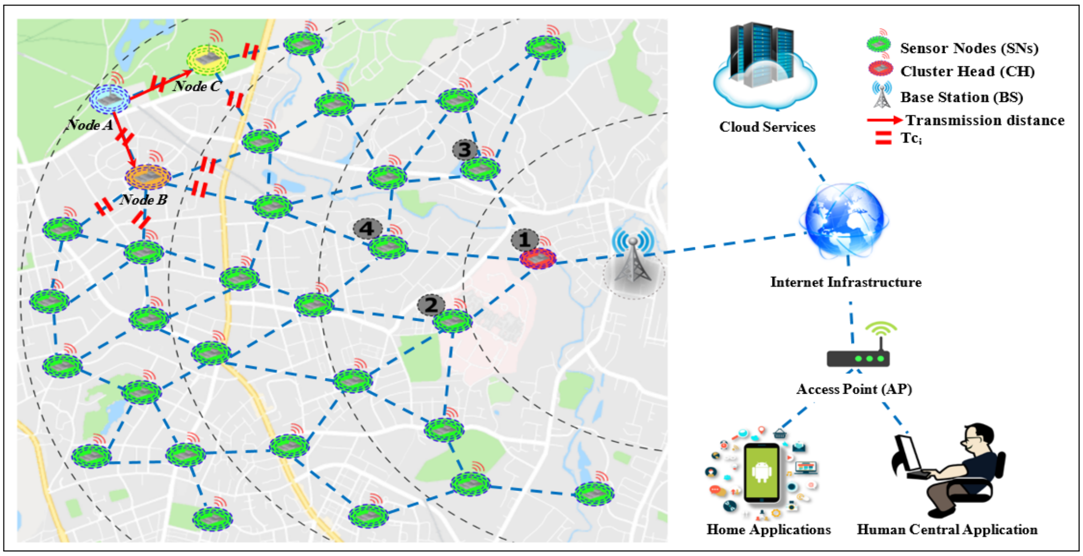


Figure 4.7 Typical WSNs clustering formation.

4.3.2 Clustering Formation

Clustering and selection of CH nodes as a strategy to minimise energy consumption for WSNs have been widely reported in the literature [166]. The CH node of a cluster acts as a coordinator within the substructure and plays a crucial role in transmitting packets. Selecting a specific node as a CH is not arbitrary. It depends on different factors and parameters, such as hop length to the BS, location of the node, its energy level, etc. Therefore, the proposed algorithm introduces a new CH selection procedure, based on tree-based clustering technique, that can reduce energy consumption.

Here, tree-based clustering uses fixed clusters to provide communication between nodes and the BS. It divides the sensing field into sub-clusters and each cluster has some number of nodes. These nodes are connected with each other using tree-based routing concepts. When a node is allocated within a cluster, it remains a member of that cluster for the entire life of the network. Thus, it eliminates the energy consumption overhead needed to select new clusters at every transmission round which is a common approach for most WSNs methods.

Shorting transmission distance between the nodes and the BS is the most energy-efficient way to deliver the data. Therefore, the proposed algorithm selects the node which has the shortest transmission distance to the BS as the CH node for its cluster.

In other words, those nodes a single hop away from the BS have higher priority when selecting the head node of a cluster. These steps are shown in Algorithm 4.1, lines 16-20.

For instance, as shown in Figure 4.7, the nodes are distributed randomly in the sensing field and they send their data to the BS through intermediate nodes. *Node* 1 is deployed as the CH node for its sub-group as it is located at the shortest distance from the BS. The nodes collect data and deliver it via intermediate nodes to this CH, and it is then forwarded to the BS. However, the amount of energy available at each node is not infinite. Therefore, when the power of *Node* 1 is exhausted, it fails to operate and is "out of service". The failure of this node should not affect the overall operation of the network, and alive node must be selected as the new CH node, based on the proposed algorithm. Thus, in the case of *Node* 1 running out of energy, the priority is given to *Node* 2, which is selected as cluster head of the sub-group, provided it has sufficient energy. *Node* 2 is selected because it is closest to the BS compared to other nodes in its sub-group. After *Node* 2 has depleted its power, *Node* 3 will become the CH of its cluster, and so on. Figure 4.8 summarises the proposed algorithm with the aid of a flowchart.

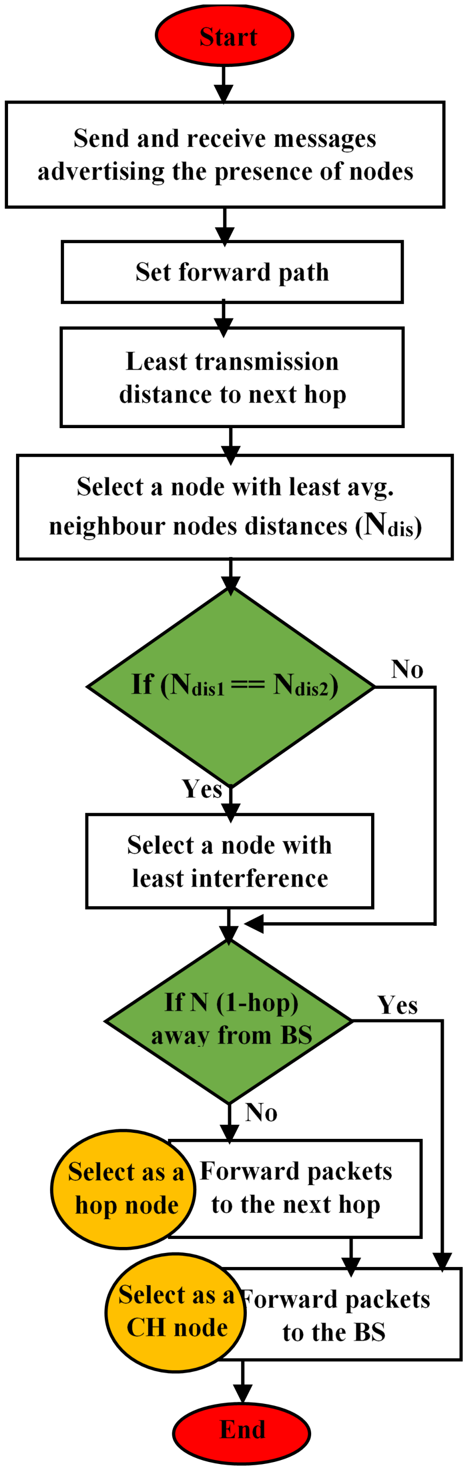


Figure 4.8 Flow-chart of the proposed algorithm.

Algorithm 4.1 :Pseudo-code for the proposed algorithm.

Initialization
Set $N_n(p)$ = neighbour nodes $(N_1(p), N_2(p), ..., etc.)$
Set SNs = sensor nodes
Set CH = cluster head
Set $d(i,BS)$ = the distance between the source to the BS
Set $min.dis$ = minimum distance to next hop
Set $Avg.dis.neig$ = average distance to neighbour nodes
Set I_n = the amount of interference for each node

procedure *Phase*1 : ROUTE DISCOVERY
  for all SNs **do**
    if $SN \in N_n(p)$ **then**
      SN send SYN packets
    else
      SN out of coverage area (in sleeping mode)
    end if
  end for
  for all $N_n(p) \in SNs$ **do**
    $N_n(p)$ *send SYN/ACK packets to SN*
  end for
end procedure

procedure *Phase*2 : ENERGY & GEO-LOCATION ROUTE
  Calculate $d(i,BS)$
  Find min.dis from (i,BS)
  if $(1-hop)$ away from the BS **then**
    A node becomes a hopping cluster member and executes *ClusterFormation*
  else
    Sensors gather the data and forward it to the next hop
  end if
  for all $SNs \in N_n(p)$ **do**
    Find min.dis to next hop (Tc_i)
    if $Avg.dis.neig_1 < Avg.dis.neig_2$ **then**
      Select $Avg.dis.neig_1$ *as the next hop*
    else if $Avg.dis.neig_1 == Avg.dis.neig_2$ **then**
      if $I_n(1) < I_n(2)$ **then**
        Select $I_n(1)$ *as the forwarder path*
      else
        $I_n(2)$ *is the next path*
      end if
    else
      Select $Avg.dis.neig_2$ *as the next hop*
    end if
  end for
  Forward packets to the target node
end procedure

4.3.3 Energy Consumption Model

A common power model [15] is assumed for this study and used to calculate the expected energy cost during the transmission and reception of data packets. It is a simple model of energy dissipation in radio hardware, where the transceiver dissipates energy to run the radio electronics and the power amplifier. The radio energy dissipation model is shown in Section 3.4, and Figure 3.10, and the total energy consumption is calculated using the equations in the same section.

4.4 Results and Discussion

This section presents the results obtained from the proposed algorithm and benchmarked against algorithms widely used today: Energy-Aware Multi-hop Multi-path Hierarchy protocol (EAMMH) [153], Power-Efficient GAthering in Sensor Information Systems (PEGASIS) [82], Tree-Based Mobile Sink (TBMS) [54].

The EAMMH scheme is one of the LEACH modification protocols which shows a new routing technique and clustering formation to deliver the data from sensor nodes to the BS. EAMMH partitions the sensing field into sub-clusters and each sub-cluster has a main-CH and number of child-CH nodes. The main-CH should be an optimum distance from these child-CH nodes. In this way, the EAMMH reduces the transmission distances between sensor nodes and the CH nodes to extend network lifetime.

PEGASIS forms chain-like links between sensor nodes to transmit the data. Each node receives the data from, and transmits it to, a close neighbour. Two nodes at the end of the chain routing structure will send the data through intermediate nodes to the leader node and then the leader transmits it to the BS. The main purpose of PEGASIS is to shorten the transmission distances between nodes, and thus the energy consumption of each node is minimised.

The TBMS technique adopts a dynamic sorting algorithm to create a tree-cluster routing structure and a mobile sink to collect the data from the sensor nodes in the sensing field. The idea of TBMS is to reduce the hop distances between nodes and thereby extend network lifetime. Further details about the EAMMH, PEGASIS and TBMS protocols are given in Section 2.4.2.1.

All the simulations were repeated 20 times using Matlab for different network topologies with the sensor nodes distributed randomly at the start of each simulation. The averages of these simulations were computed and the obtained results are shown below. All the parameters used in the simulations are listed in Table 4.1 [54]. Additionally, we considered four different scenarios for the validation of the effectiveness of the proposed scheme, see Table 4.2.

Figure 4.9 shows the average transmission distances for the four schemes based on the scenarios presented in Table 4.2; where the average transmission distance is the total transmission distances for all sensor nodes divided by the number of sensor nodes. We see the lower transmission distances are best for more efficient data dissemination. It is obvious from Figure 4.9 that the proposed algorithm gave lower transmission distances than other three for all scenarios considered.

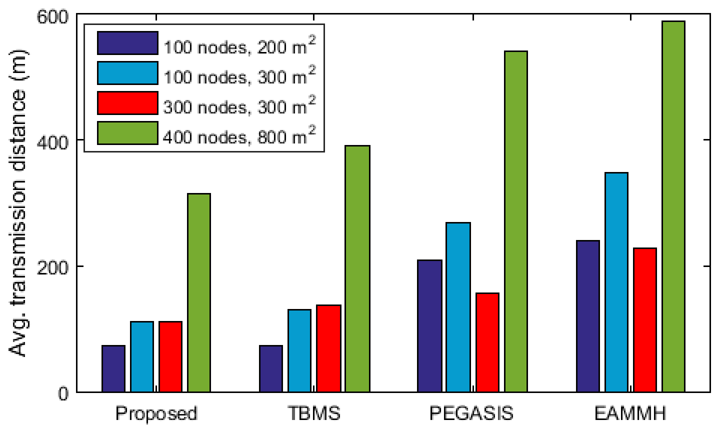


Figure 4.9 Avg. transmission distance (m).

Figure 4.10 presents the average number of hops for each of the four algorithms based on the scenarios shown in Table 4.2. The average number of hops is the summation of data relays in multi-hop communication to reach the BS, divided by the number of nodes in each scenario. Clearly, the EAMMH protocol has fewer hops than the other algorithms. This is because of the multi-hop transmission behaviour used in this approach; the node sends its data in one or two long hops to the BS. Conversely, the PEGASIS protocol has a highest number of hops, which reduces end-to-end reliability and the lifetime of the network. However, the optimal number of hops is achieved by both proposed protocol and TBMS technique.

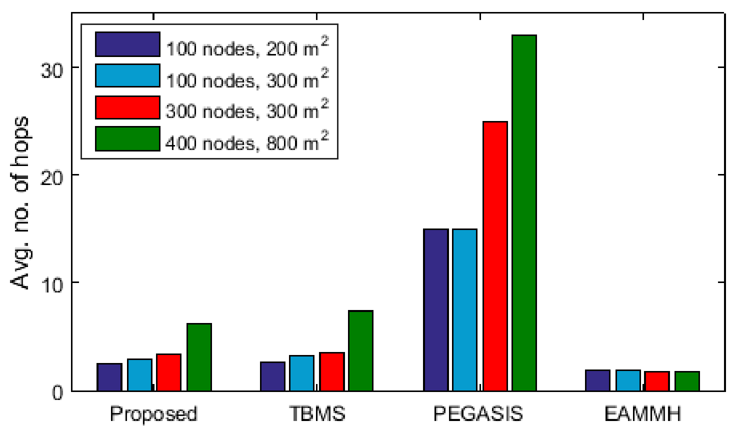


Figure 4.10 Avg. number of hops.

Figure 4.11 demonstrates the average energy consumption for the first transmission for the four algorithms based on scenarios of Table 4.2. The energy was calculated for each node by including the energy used for collecting, transmitting, advertising and receiving packets. It is clear that the average energy consumption for all four algorithms increases when the number of nodes and sensing area increased. Clearly, in accordance with Figure 4.9, the proposed scheme has lowest transmission distance, and this leads to lower energy consumption than the EAMMH, PEGASIS and TBMS algorithms by 64%, 62%, and 29%, respectively, in the case of the largest number of nodes (400) and the greatest area (800 m^2) covered.

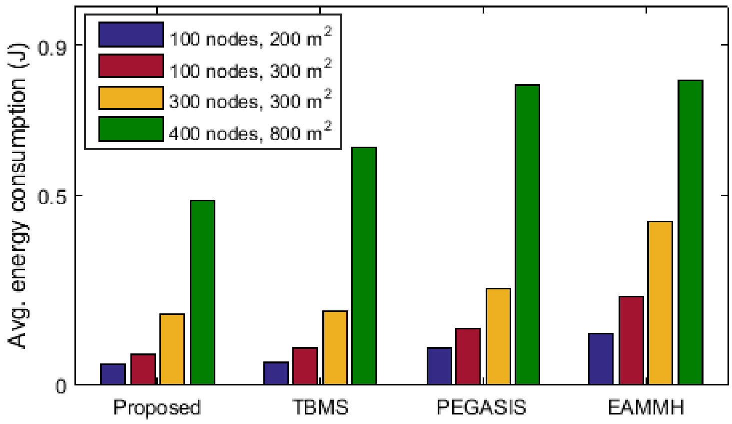


Figure 4.11 Avg. energy consumption for first transmissions (J).

Table 4.3 shows the simulated network lifetime for the four protocols, as determined by the death of the last node. The results reveal that the proposed scheme has the longest lifespan for all the scenarios considered. This is consistent with the results presented in Figures 4.9 and 4.11.

Table 4.3 Network lifetime till the last node dies.

Nodes and sensing area	Proposed	TBMS	PEGASIS	EAMMH
100 nodes, 200 m^2	594	594	396	262
100 nodes, 300 m^2	562	533	317	241
300 nodes, 300 m^2	605	417	392	263
400 nodes, 800 m^2	447	313	247	161

Figure 4.12 shows total energy remaining in the network after each round when the sensing field had an area of 200 m^2 with 100 nodes. It appears that TBMS and the proposed protocol give almost the same values. This means that a random mobility pattern is suitable for fewer nodes and smaller area (100 nodes and 200 m^2). The TBMS scheme and the proposed method outperform other protocols.

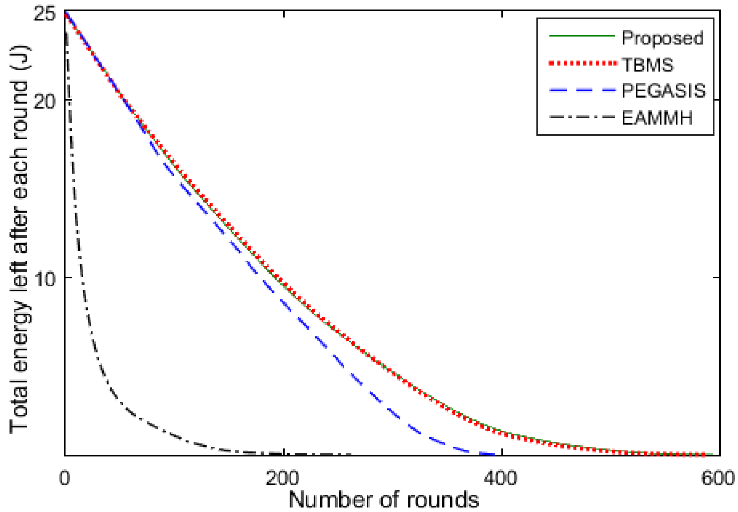


Figure 4.12 Total remaining network energy (200 m^2 sensing area and 100 nodes).

To prove that our protocol was well-designed, the sensing area and number of nodes were extended as in Table 4.2. It appears from Figures 4.13 to 4.15 that the proposed technique achieves better energy savings than the EAMMH, PEGASIS and TBMS algorithms, by as much as 177%, 80% and 42% respectively in the case of the largest number of nodes (400) and the greatest area (800 m^2) covered.

The current protocol performs better as the number of nodes and sensing area increase. This is because the proposed protocol takes a smart decision to balance the traffic load; it shifts the traffic from overloaded nodes to other nodes with less traffic, and thereby reduces the network congestion. This leads to an increase in the lifespan of the nodes. Figures 4.13 to 4.15, show that the proposed algorithm significantly improved network lifetime compared to EAMMH, PEGASIS and TBMS methods.

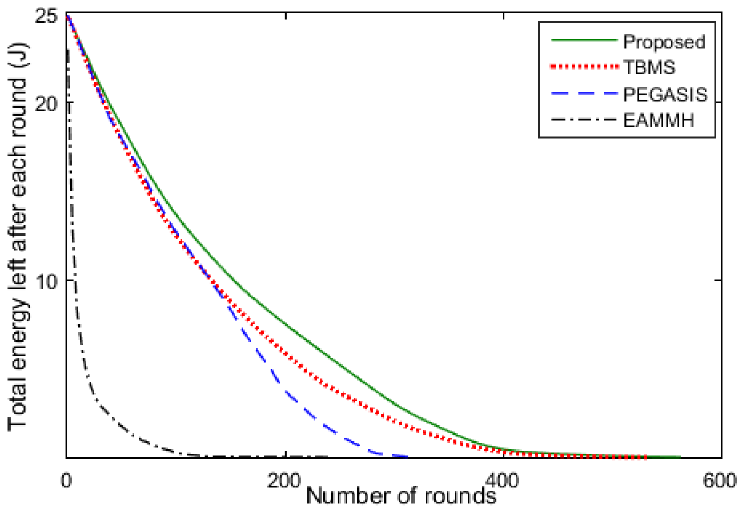


Figure 4.13 Total remaining network energy (300 m^2 sensing area and 100 nodes)

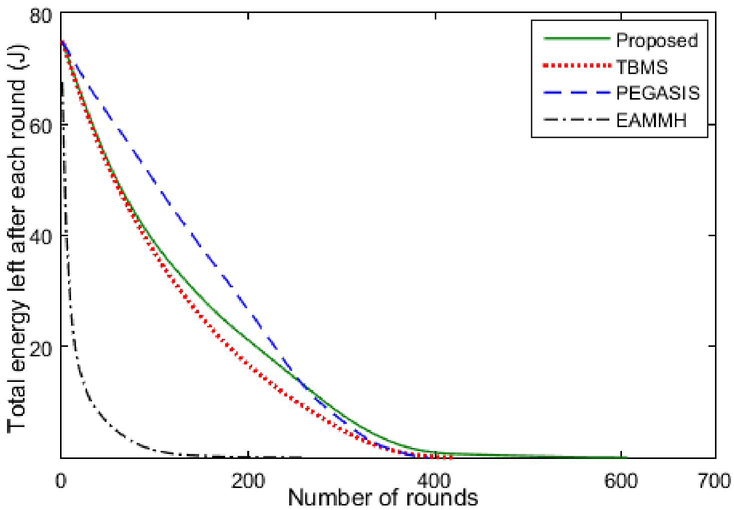


Figure 4.14 Total remaining network energy (300 m^2 sensing area and 300 nodes)

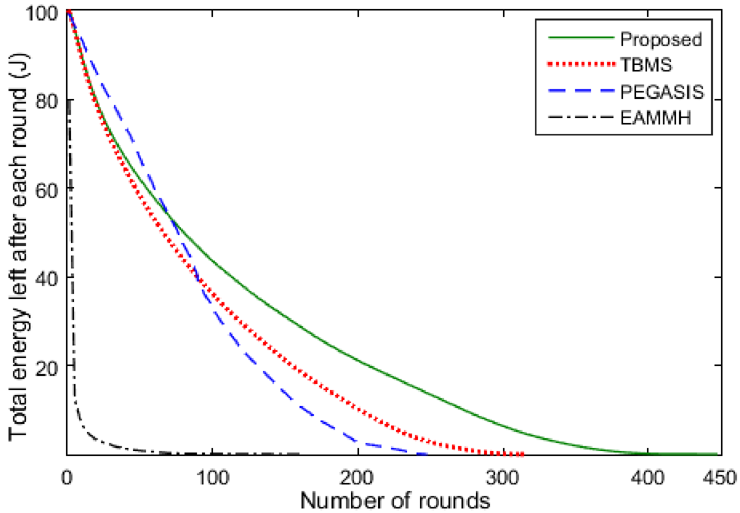


Figure 4.15 Total remaining network energy (800 m^2 sensing area and 400 nodes)

The simulation results shown in Figures 4.12 to 4.15 are summarised in Table 4.4, together with results obtained for the other scenarios shown in Table 4.2 which shows that the proposed algorithm outperformed the other three. This is because the proposed algorithm finds a better load balance between the nodes in the sensing field and thus extends network lifetime.

Table 4.4 Relative improvement in network lifetime obtained with proposed algorithm compared to EAMMH, PEGASIS, and TBMS.

Nodes and sensing area	TBMS	PEGASIS	EAMMH
100 nodes, 200 m^2	0%	50%	126%
100 nodes, 300 m^2	5.4%	77%	133%
300 nodes, 300 m^2	45%	54%	130%
400 nodes, 800 m^2	42%	80%	177%

The sets of results below show how the proposed algorithm compared to the other three in terms of nodes remaining active, delay time, the number of retransmissions per each message (RPEM), transmission overhead and throughput based on Table 4.2 scenario 4.

Figure 4.16 reveals the number of alive nodes remaining after a given number of rounds for each of the four algorithms. It is seen from the figure that the proposed scheme achieves a longer network life than the others. The proposed algorithm extends the network lifetime compared to EAMMH, PEGASIS and TBMS by 177%, 80% and 42% respectively.

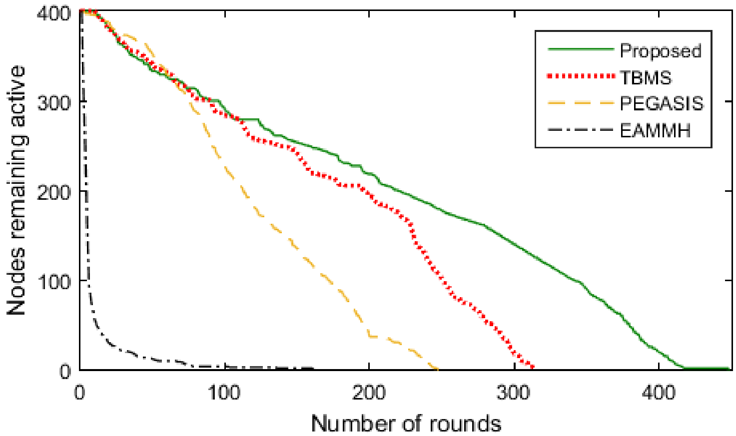


Figure 4.16 Nodes remaining active after each round for each of the four algorithms (EAMMH, PEGASIS, TBMS and the one being proposed here).

Figure 4.17 depicts the average transmission time for the all the algorithms considered based on Table 4.2 (scenario 4). The transmission time is the time taken by the packet to travel through the communication medium from the source node to the ultimate receiver. During the multi-hop transmission process, each sensor requires the processing time to send and receive data. The simulation setting adopted was based on [54, 154], which took 2 ms for a sensor node to make a transmission. All sensor nodes updated their information each interval period. The length of an interval for update packets is 200 ms. The propagation delay is calculated by: ($transmission\ distance$ / $speed\ of\ light$). Therefore, the total transmission time is: ($propagation\ delay + processing\ time\ for\ each\ transmission$). It appears from this figure that the average delay time for the proposed scheme is lower than for EAMMH, PEGASIS and TBMS. This is a clear trend of decreasing transmission distances by the proposed algorithm.

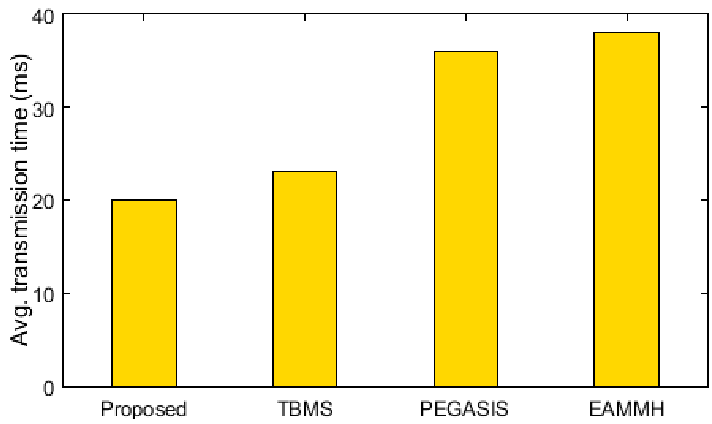


Figure 4.17 Average transmission time (s) for each of the four algorithms (EAMMH, PEGASIS, TBMS and the one being proposed here).

When a hop node or CH dies due to energy exhaustion, this means a packet is retransmitted and requires the sensor nodes to send data to the BS through the hop node or the CH node. Figure 4.18 depicts the RPEM for the four algorithms, for 400 nodes and a sensing area of 800 m^2. It is clear that the proposed scheme has a lower number of RPEM than the others. This means that most messages can be delivered successfully per round. In other words, the throughput is higher than for the other three schemes. It reflects that the proposed algorithm brings a lower transmission overhead for each round as can be seen in Figure 4.19.

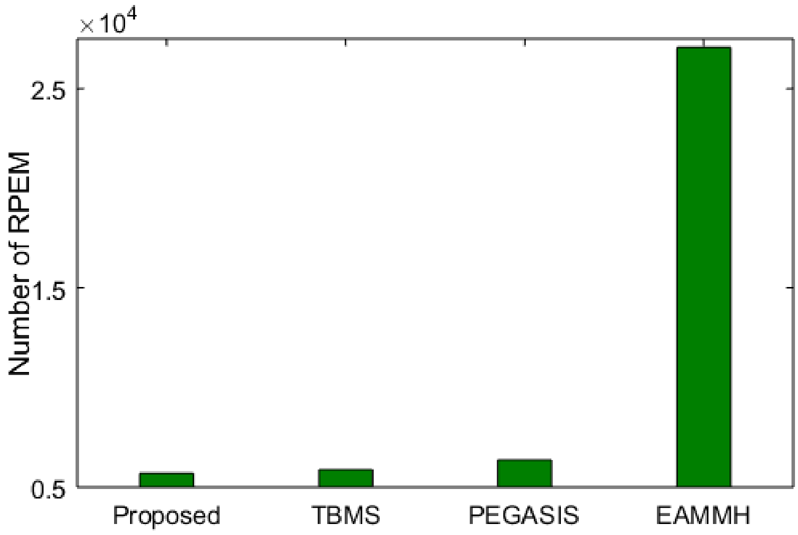


Figure 4.18 Number of RPEM for each of the four algorithms (EAMMH, PEGASIS, TBMS and the one being proposed here).

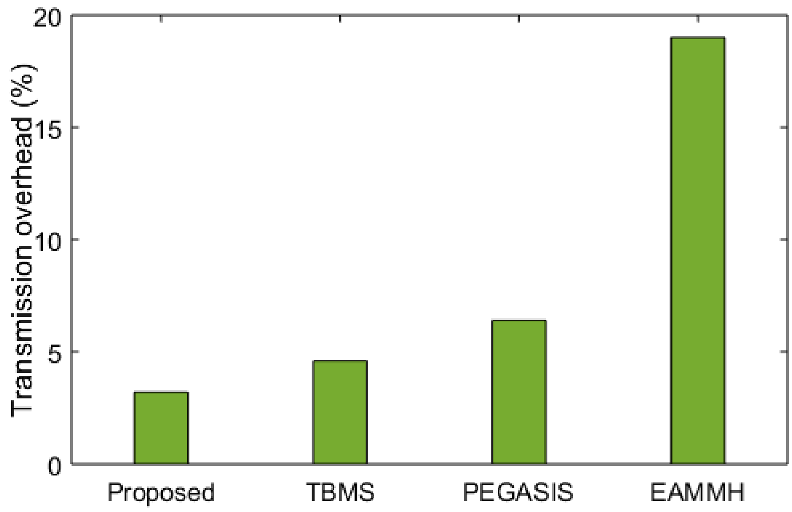


Figure 4.19 Transmission overhead (%) for each of the four algorithms (EAMMH, PEGASIS, TBMS and the one being proposed here).

Figure 4.20 shows the throughput of the proposed algorithm compared to the three alternatives, where the throughput is the percentage of successfully delivered packets sent from sensor nodes to the BS each round. As can be seen from the figure, the proposed scheme has higher throughput because of the optimal number of hops and low RPEM during the data transmission process.

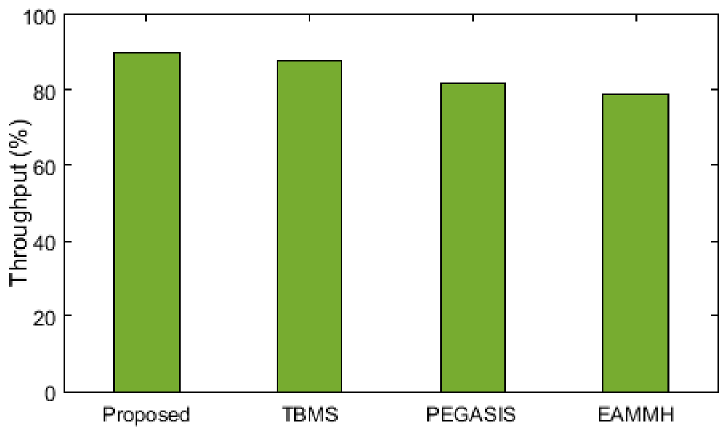


Figure 4.20 Throughput (%) for each of the four algorithms (EAMMH, PEGASIS, TBMS and the one being proposed here).

4.5 Summary and Contributions to Knowledge

In order to minimise energy consumption and enhance WSN lifetime, a new routing strategy and distributed clustering formation have been presented in this chapter. It has been demonstrated that larger numbers neighbour nodes has a negative impact on the overall network in terms of energy consumption and communication traffic. This investigation demonstrated that the transmission overhead and energy consumption increased when the number of neighbour nodes increased. To minimise energy consumption and extend network lifetime, the proposed method of CH node selection is founded on tree-based clustering and a low transmission distance between the CH node and the BS. The performance of the proposed algorithm was evaluated in terms of power consumption, end-to-end delay and transmission distances, and then compared with EAMMH, PEGASIS and TBMS. Detailed analysis showed that the proposed scheme achieved improvements of 64%, 62%, and 29% in energy conservation as compared to EAMMH, PEGASIS and TBMS protocols, respectively. The proposed scheme also extended the network lifetime by 177%, 80% and 42%, respectively relative to EAMMH, PEGASIS, and TBMS.

The major contributions of this chapter are summarised as:

- Some mathematical equations of interference and number of neighbour nodes, and their relationship with transmitting power has been developed. It shows that increasing the number of neighbour nodes increases interference, and the transmission power.

- Based on the number of neighbour nodes, the estimated interference for each sensor node and less transmission distances, the proposed algorithm makes intelligent decisions to forward the data to the next-hop node when determining routing.

- An important feature of the proposed technique is well-balanced network traffic minimising the probability of packet collisions and so increasing the performance of the overall network.

- A new clustering architecture using the multi-hop concept and which reduces transmission distances has been adopted to reduce the energy consumed by each node and so extend the network lifetime.

- The proposed protocol combined energy considerations with geo-location routing to select a path with less interference.

The next chapter focus on an efficient data transmission and a real-time remote monitoring system that reduce the energy consumption for WSNs and IoT devices.

Chapter 5

Efficient Data Transmission and Remote Monitoring System for IoT Application

Aggregation and transmission of data are considered as the main reasons for energy consumption by WSNs and IoT devices. These devices waste some of their energy in processing and transmitting redundant and unnecessary data. Therefore, this chapter presents means to eliminate redundant data and consequently reduce the number of data transmissions. As a result, energy consumption of the IoT device is reduced. The study also provides a remote monitoring system for the end-user that can check and track the performance of these sensors/IoT devices during real-time communication.

5.1 Introduction

As explained earlier the uses of IoT and WSNs often require a dense deployment of sensor nodes over the area to be monitored [46]. Dense deployment will often produce highly related and redundant data which need to be processed for redundancy checks and routed appropriately in a single or multi-hop manner to the ultimate receiver [72].

Sensor nodes normally rely on limited battery power and each sensor is composed of three main parts, sensing unit, computing unit and communication unit [129]. All three units require power for operation. According to an investigation by [13], the communication unit uses 60% of the energy available, and the sensing unit consumes 30%. As a result, transmitting multiple copies of the same packet consumes proportionally more energy and leads to reduced the network lifetime. Thus, adopting a suitable aggregation method that allows the connected devices to sense useful data and share only these data when required is highly recommended.

In this chapter, as an example of the many industries that use real-time networks to promote the operation of manufacturing processes [167]. We choose a gas turbine engine blade as an applications of the IoT to sense, forward and monitor data over the internet, based on the cloud infrastructure. The objective of this chapter is to filter and eliminate redundant data, reduce the number of packets transmitted, and decrease the amount of data transmitted over the network. This should improve energy efficiency and performance of the entire system. It also proposes a real-time monitoring system that allows the end-user to interrogate the transmitted data and quickly check the machine's condition in real-time.

In the previous chapters, the study focused on scheduling and routing algorithms for WSNs that reduced energy consumption and extended network life. The algorithms were developed for a futuristic IoT paradigm where a large number of nodes are distributed over a large geographical area. In this chapter, we show a real-time application of a smaller scale use of IoT on an industrial setup. We focus on the different but equally important problem of data redundancy which increases energy consumption. The aim of this chapter is not to validate the algorithms from previous chapters on a hardware platform, but to demonstrate a real application of how data should be handled by constrained IoT devices and transmitted from these to the ultimate receiver over the internet. This leads to a reduced volume of data transmissions and thereby reduces the process of scheduling and routing data on each device in the network. As a result, this also acts to reduce energy consumption and foster the success of IoT technology.

5.2 Network Configuration

This section provides a description of a suitable aggregation technique to reduce the data redundancy of a typical IoT monitoring system. A distributed software and hardware system are developed that can sense the analogue data via sensors from the blade of the gas turbine engine and forward it to a Raspberry Pi (RasPi) through a Custard Pi (CPi). The RasPi is one of the key learning platforms for IoT. It provides a complete operating systems (i.e., Raspbian Linux, Windows, Android, , IoT Core, etc.) in a tiny platform for a very low cost [168], and was used as an IoT device for this study.

However, RasPi does not have any analogue input to read the data from sensors [169] and the CPi was used to convert the data from analogue to digital. After that, the RasPi processed and filtered data redundancy and then forwarded digital information to the end-user via a cloud infrastructure over an IP network. The use of the RasPi should, in effect, reduce the overall usage of resources and improve the efficiency of the network. The functions of the various components and software are explained below and shown in Figure 5.1.

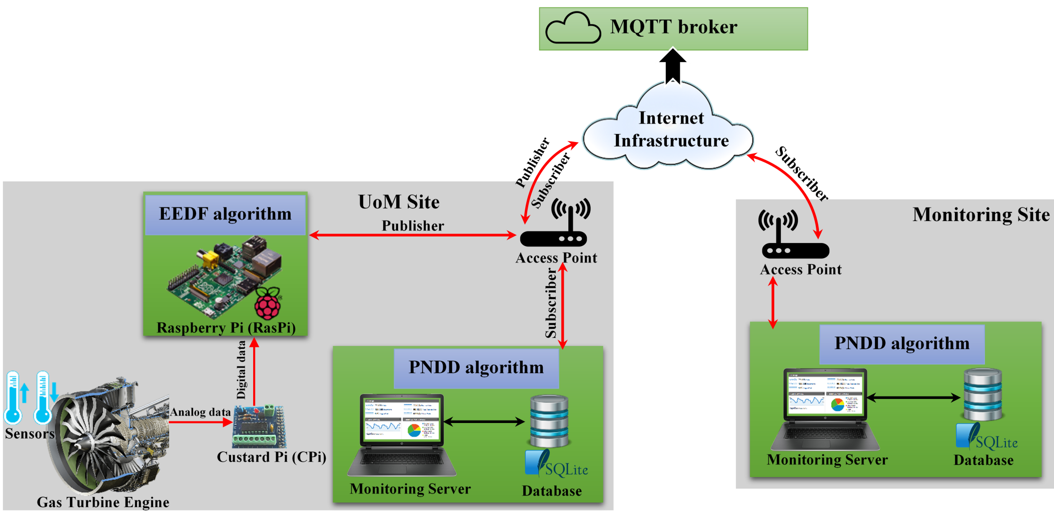


Figure 5.1 Overview of system architecture.

The overall system is divided into two main parts. The first part is located at the University of Manchester (UoM). It includes the test section (water rig) which simulates the blade of a gas turbine engine. The reason for using water as the working fluid is that due to the lower kinematic viscosity of water compared to air, high Reynolds and rotation numbers can be achieved at lower velocities [170]. The test section had six sensors positioned randomly in different locations of the test section to detect the flow rate of the water, and provide data to the publisher device (RasPi) through the CPi board. The RasPi collects and processes these data based on the energy-efficient data forward (EEDF) policy and then transmits to the hosted message broker (MQTT) over the internet.

The second part of the system is located at the monitoring site and contains the subscriber device (monitoring server) and database to receive the data that comes from the MQTT broker, and to store it on the monitoring server database using the predictive non-dispatch data (PNDD) algorithm. The PNDD protocol also provides a real-time plotting system to the end-user on the monitoring server.

Figure 5.2 shows the data flow direction in each part of the system. The proposed system consists of several subsystems and protocols that manipulate the data to share and communicate with each other, each will be explained in detail below.

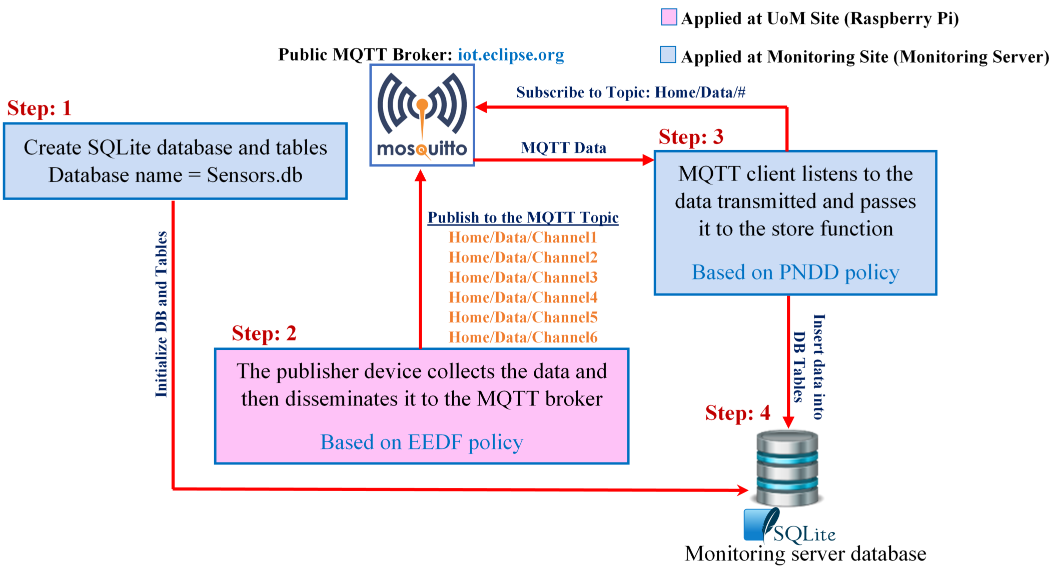


Figure 5.2 Schematic diagram of the proposed system.

5.2.1 Message Queuing Telemetry Transport (MQTT) protocol

MQTT is one of the most commonly used protocols in IoT applications. [32]. It was originally created and developed by the IBM company [171]. It is an ISO standard based on the publish/subscribe (pub/sub) communication pattern [172]. MQTT works on the top of the TCP/IP protocol which can carry a sequence of nearly 256 megabytes of data, usually adding a fixed header of two bytes to most payloads [135]. The main purpose of this protocol is to reduce the burden of IoT constrained devices for sending/receiving messages. These devices usually have limited memory, power, bandwidth and processing unit.

There are three kinds of actors in the pub/sub architecture system; publisher, broker and subscriber [173]. A publisher device sends the messages identified by a specific topic to the broker. Then, the broker forwards these messages to every subscriber device interested in that topic. A broker handles all messages passing from the publisher to subscriber devices. There are no standard topics for the MQTT protocol. The publisher can define any topic with one or more subscribers.

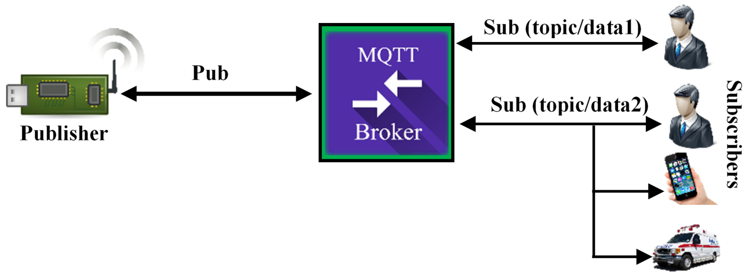
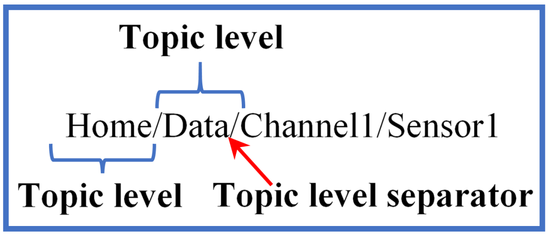


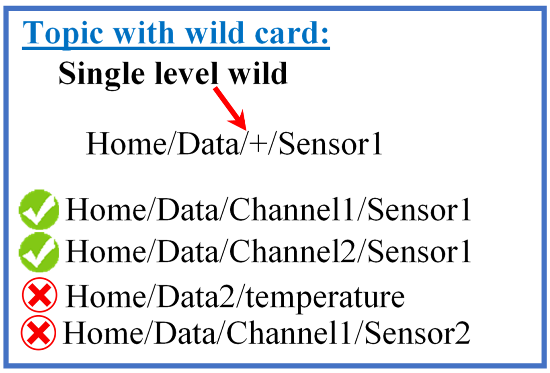
Figure 5.3 An example of MQTT protocol use-case with different topics and clients.

MQTT protocol supports a hierarchical topic name-space [12]. The subscription topics consists of one or more topic levels. Each level is separated by a forward slash "/" as shown below:



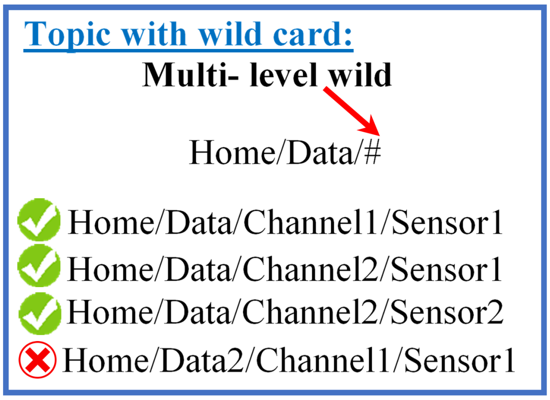
Two wild cards characters are supported for subscriptions to hierarchical topics as described below [174]:

– Single Level Wild-card (+): presents one topic level of the hierarchy and uses the character "+" between delimiters, for instance:



– Multi-Level Wild-card (#): covers many topic levels. The character # represents the multi-level wild card in the topic. It must be placed as the last character in the topic and is preceded by the character "/".

The MQTT protocol supports three possible quality of service (QoS) values concerning the guarantee of delivering a message between publisher and subscriber devices. The QoS level implementations for the MQTT protocol are [174]:



- **QoS (0), at most once:** is the simplest, fastest mode method of transfer with lowest overhead. The sender simply publishes the message to the broker, there are no acknowledgement packets between the publisher and the broker. Therefore, there is no guarantee of message delivery.

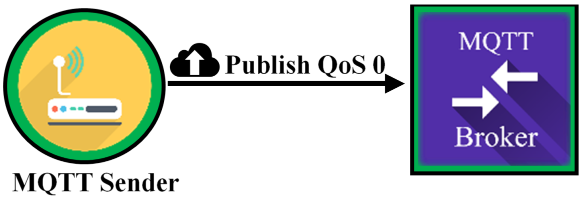


Figure 5.4 Quality of Service (0) of MQTT.

- **QoS (1), at least once:** delivers the messages at least once to the receiver. The sender forwards the message and waits for an acknowledgement (PUBACK) packet from the broker. This level guarantees that the message will be forwarded successfully to the broker. However, the message may reach the broker more than once.

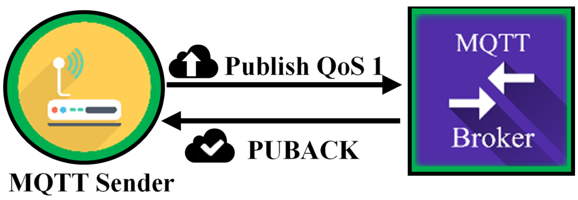


Figure 5.5 Quality of Service (1) of MQTT.

- **QoS (2), exactly once:** is the highest level of MQTT services. This level guarantees that the message will be received once by the broker. There is a sequence of four messages (a four-part handshake) between the sender and MQTT broker. When the handshake messages have been completed, both the sender and receiver have confirmed that the data was sent exactly once.

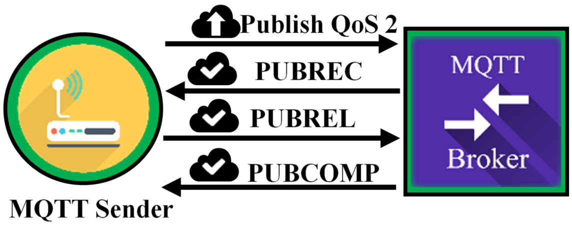


Figure 5.6 Quality of Service (2) of MQTT.

Because the MQTT protocol consumes less power and is faster than the HTTP, many systems use the MQTT instead of the HTTP protocol as the core of data transportation [175]. Therefore, in this study, the MQTT protocol has been selected to control the IoT device (RasPi) through the internet. Furthermore, the experiment results in [172] showed that the QoS (2) has a lower packets loss rate than QoS (0) and QoS (1). Therefore, the QoS (2) was selected for this study in order to receive the data exactly once.

Most MQTT brokers do not have any built-in technique to store MQTT client data permanently into the database. If the connection between sender and broker is QoS (0) then the data does not need to be stored, and the messages are delivered to only the currently active and connected subscribers [176]. When the connection is QoS (1), the messages often have to be stored temporarily on the broker if the subscriber is online. However, the message is stored for longer if the subscriber is offline or the retain flag is on the message [177]. However, if QoS (2) is selected then the messages are stored in the broker for a period of time until the transaction is fully committed [177]. For these reasons, a database must be installed on the monitoring server to store the data sent by the publisher device.

5.2.2 Embedded Database SQLite

SQLite is an open-source and extremely lightweight relational database considered the most widely used of all database engines [178]. It was written by D. Richard in 2000 using C-language [179]. SQLite is used by several web browsers in common use (e.g., Safari), in embedded systems (e.g., mobile phones), and operating systems (e.g., Windows 10). It is faster than other popular client/server SQL database engines such as MySQL, Oracle, etc. for most common operations [180]. SQLite is a zero-configuration which means no complex setup or administration is needed and it supports terabyte-size databases [181]. Therefore, it was selected as an embedded database for monitoring server storage.

5.2.3 Eclipse Paho Library

"Paho project was created to provide scalable open-source client implementations of open and standard message protocols aimed at new, existing, and emerging applications for IoT [182]". Paho initially started with MQTT publish/subscribe client implementations for use on embedded platforms. Paho client libraries are available on various platforms and programming languages such as Python, Java, C, C++, etc. [182]. For this study, Paho client programming is performed in Python.

5.2.4 Raspberry Pi Single Board Computer

RasPi is a tiny and affordable microcomputer based on the Raspbian Linux operating system [133]. It comes in different models ranging from RasPi1 A (first produced in 2013) to RasPi4 B (first produced in 2019). This study used model RasPi3 B because it is cheaper than other models and still suitable for our purpose. It includes 512 megabytes of RAM and an ARM1176JZF-S 700 MHz processor [168].

Figure 5.7 shows the RasPi board. The board does not include a built-in hard disk and so relies on a micro secure digital (SD) card for booting and long-term storage. It also uses a 5 Volt power supply with recommended input current of 700 mA. The unit offers less complexity and better solutions than other IoT platforms for monitoring IoT devices [183]. In this research, the RasPi is used as an example of an IoT device to collect the data from six sensors deployed in the test section, then to process the data, and send the data to the end-user (monitoring server).

The output of these sensors are voltages, and the RasPi converts the output data into pressure using the digital manometer and Equation 5.1. For a given voltage (V, milli-volts), the pressure (p, mbar) is given by:

$$Pressure(P) = (261.95 * V) - 129.96 \tag{5.1}$$

Where V is the aggregated data from a single sensor. The RasPi was integrated with Wi-Fi, ethernet port and Adafruit IO as an IoT platform. It also included USB ports, general purpose input/output (GPIO) pins to interface with LEDs, high definition multimedia interface (HDMI) port, and others. The digital manometer device includes calibration facilities and can be used to measure low-pressures [184].

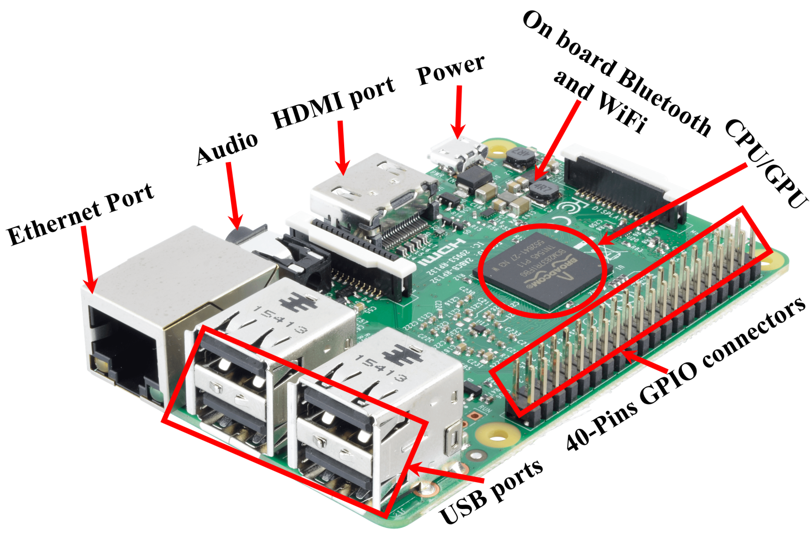


Figure 5.7 Raspberry Pi board (model RasPi B).

5.2.5 Custard Pi Add-on Board

The CPi board is a range of cards that connect to the GPIO pins of the RasPi to provide a variety of options [185]. The RasPi GPIO does not support any analogue inputs and thus, CPi was added. The CPi has 8 analogue channels with 12-bit analogue-to-digital (ADC). The CPi 1 and 2 use non-stackable connectors. Hence, once the board is fitted into the RasPi, nothing else can be plugged into the GPIO connectors. However, the CPi3, 4 and 5 does use stackable connectors which means that other boards can access the GPIO pins if required. CPi 6 uses a ribbon cable to transfer the data and is too large to plug into the RasPi [185]. The current study used CPi 3A. The board simply plugs into the 26-way (RasPi B) GPIO connector. The LED is fitted on the board and supplied with the 3.3 V to confirm correct plug-in. The RasPi is protected from accidental connection of a high voltage by the use of an analogue to digital chip connected to the SPI bus using MCP3208 IC. Each analogue input is 2.5 V per channel in single-ended mode. Easy to connect crew terminals are supplied for the 8 channels for the external sensors. The CPi3A unit is shown below:

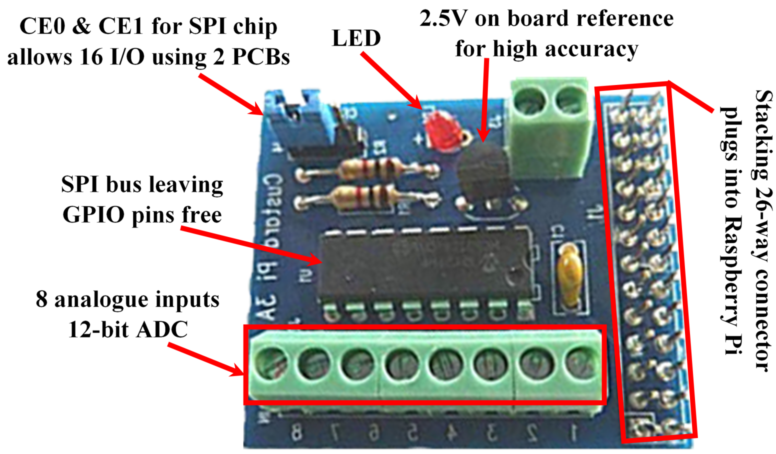


Figure 5.8 Custard Pi 3A.

5.2.6 Pressure Transmitter (Type 663)

The Huba Control Type 663 pressure level transmitter, see Figure 5.9, has calibrated and temperature compensated sensor signals output as a voltage [186]. Sensor sensitivity/accuracy is largely unaffected by temperature, because each sensor automatically compensates for surrounding temperatures in range -20 to 70 °C [186]. The pre-amplified linear voltage output signal can be connected directly into an electronic control system. This study used six pressure transmitter sensors to collect data from the test section. Each sensor is positioned in a different place in the test section and connected to an individual channel via the CPi board. The pressure transmitters read the pressure values and forwards them to the RasPi via the CPi.

Figure 5.9 Huba type 663 differential pressure transmitter.

5.3 Data Filtering and Predicting Processes

This study aims to reduce redundant data transmissions and thus help minimise energy consumption of the IoT device (RasPi), to help minimises transmission delay, amount of data and bandwidth used over the entire network (i.e., from publisher to the subscriber device). The study also provides a real-time monitoring system to the end-user that allows checking and tracking of the status of the device/sensors. In such a study, the data aggregation is generally divided into two stages. The first stage is called the energy-efficient data forward (EEDF) algorithm. While the second stage is called the predicted non-dispatch data (PNDD) algorithm. The EEDF protocol senses, filters and forwards the data by RasPi to the MQTT broker. However, the PNDD protocol predicts and receives the data from the MQTT broker and then stores and plots these data on the monitoring server. The EEDF and PNDD algorithms were written in Python and implemented in the RasPi and monitoring server, respectively.

In the following sub-sections, further details about the proposed algorithms are described by highlighting their techniques and strategies.

5.3.1 Filtering Process

The first purpose of the EEDF algorithm is to sense the analogue data from the six sensors connected to the RasPi via the CPi. The RasPi is required to establish a connection with the MQTT broker before the actual data communication begins. Any device that is connected to the network and exchanges application messages through the MQTT broker requires an MQTT client. Therefore, an MQTT client is installed on both publisher (RasPi) and subscriber devices (monitoring server) to connect, disconnect, publish and subscribe data. The RasPi must specify a topic to publish data to the MQTT broker, and only the device that is subscribing to the same specified topic can receive that data from the MQTT broker. By establishing connections between the RasPi and the MQTT broker based on MQTT QoS (2), the important decisions taken are based on the EEDF algorithm.

Each sensor senses 7 bytes of data, obtained by sensing events from the test section. Thus, the total data gathered by all sensors is 42 bytes. It is now proposed that the EEDF algorithm also checks and compares simultaneously collected data with the previous record for each sensor individually. If the collected data in each sensor is not equal to the previous record, then it is sent to the broker via the internet. Otherwise, the EEDF protocol detects no change in the data and then filters the duplicated data out and thus this data is not sent to the broker. As a result, the RasPi antenna goes into sleep mode. Of course, in the first round, all the collected data is sent to the broker because there is no previous data record for comparison. As a result, the proposed scheme minimises the processing and transmission time for the RasPi. This, in turn, helps minimise the energy consumption of the RasPi. It also reduces the delay time and bandwidth used over the entire path of the network and improves the performance of the end-to-end system. The flowchart of the proposed algorithm to sense and forward the data is presented in Figure 5.10.

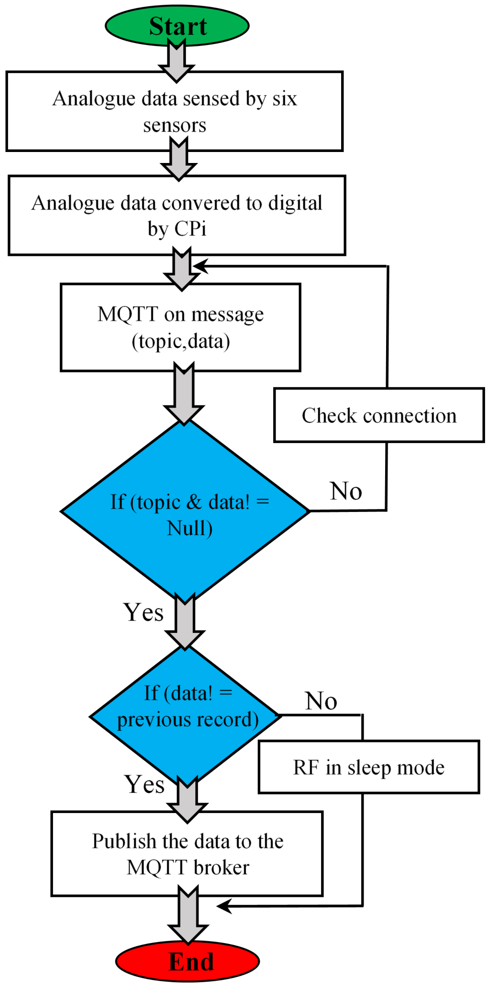


Figure 5.10 Flow chart of the publisher algorithm.

5.3.2 Predicting Process

The first concern of the PNDD protocol is to establish the connection between the monitoring server and MQTT broker based on MQTT QoS (2). When the QoS (2) process is complete, the connection is created and the monitoring server can communicate with the MQTT broker and receive messages. However, the PNDD algorithm requires a database to store the data that comes from the MQTT broker. Therefore, SQLite was installed on the monitoring server to store, analyse and plot the data.

After that, the PNDD protocol listens to the MQTT broker and gets a copy of the data forwarded by RasPi. The transmitted data could be a number of bytes or an empty row based on the EEDF publisher policy. The PNDD algorithm receives and stores the empty row of data when there is a connection between monitoring server and RasPi through the MQTT broker, but no data is received (data filtered by EEDF algorithm). Otherwise, the PNDD algorithm stores the transmitted data (payload) in the monitoring server database.

Real-time plotting plays a significant role in the diagnostics of any remote monitoring system. Thus, the PNDD protocol provides the end-user with a real-time plotting system for further processing and diagnosing the information. The PNDD algorithm predicts and retrieves data when it is required. If the collected data in the database is not equal to the empty row data, then it plots the current data. Otherwise, it plots the previous record as the new data. As a result, the PNDD protocol reduces the capacity required to store data. The flowchart of the PNDD algorithm is shown in Figure 5.11.

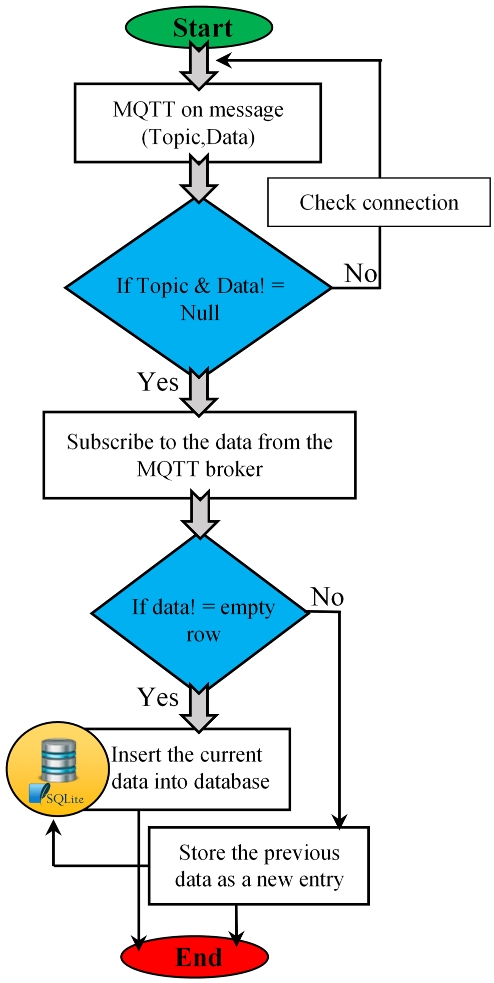


Figure 5.11 Flow chart of the subscriber algorithm.

5.3.3 Remote Monitoring Systems

IoT has gained considerable popularity and is widely applied in many fields, but different devices using various sensors are utilised for monitoring activities [3]. These sensors collect data and use that data to trigger automatic alerts and actions such as maintenance requests, remote diagnostics and other optional processes [187].

Thus, this chapter, in addition to a filtering mechanism also proposes a remote monitoring system for a gas turbine engine in an IoT environment. The system aims to monitor and diagnose the condition of the gas turbine engine, sensors and IoT device (RasPi). Here, the RasPi acts as the gateway, connecting the sensors to the internet. The data is gathered by the six sensors and forwarded to the RasPi which uses its WiFi connectivity to establish the connection and disseminate the data to the monitoring server (end-user) based on the MQTT broker.

On the monitoring side, each item of data that is sent by each sensor requires the plotting of an individual sub-figure. Therefore, the end-user has a dashboard for displaying and analysing the data for each sensor (see Figures 5.14 to 5.19). Based on these figures, the end-user can check the system status, sensors or RasPi for errors during real-time communication. For instance, the end-user checks the engine condition based on the x- and y-axes of the sub-figures. Should the RasPi experience connectivity issues or sensor stop sending data to the end-user, the sub-figure stops plotting the data. Thus, the end-user can see which sensor is faulty. Figure 5.12 shows the architecture of the proposed system which consists of four main parts: engine, sensors, gateway (RasPi) and end-user.

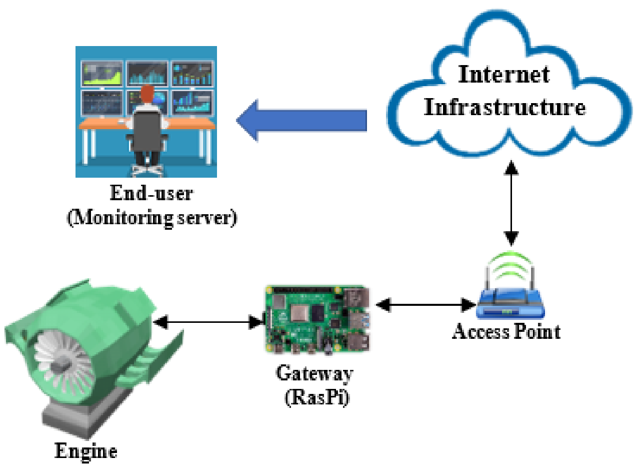


Figure 5.12 Remote monitoring system for gas turbine engine using IoT.

5.4 Experimental Setup

This section presents the experimental setup to evaluate the performance of the proposed algorithms for an IoT application. The hardware is shown in Figure 5.13. Water is used as the working medium and pumped into the test section from the tank. Measurement of the volume flow rate of the water is carried out by six transmitter sensors. The sensors are interfaced and powered by the CPi. They are placed in different positions in the test section to sense and forward the data to the RasPi via the CPi. The CPi is linked to and powered by the RasPi. The RasPi is used to process and forward the data to the monitoring server via the cloud. The data is dispatched by the RasPi every 15 second. The network was tested for twenty-five minutes. The parameters and protocols used in the setup are shown in Table 5.1. Here the RasPi is mains powered, but in a real situation, it will most certainly be battery powered, i.e., resource-limited. To validate the reduced energy consumption, the measurements were made using a USB 3.0 LCD voltmeter ammeter voltage multimeter tester.

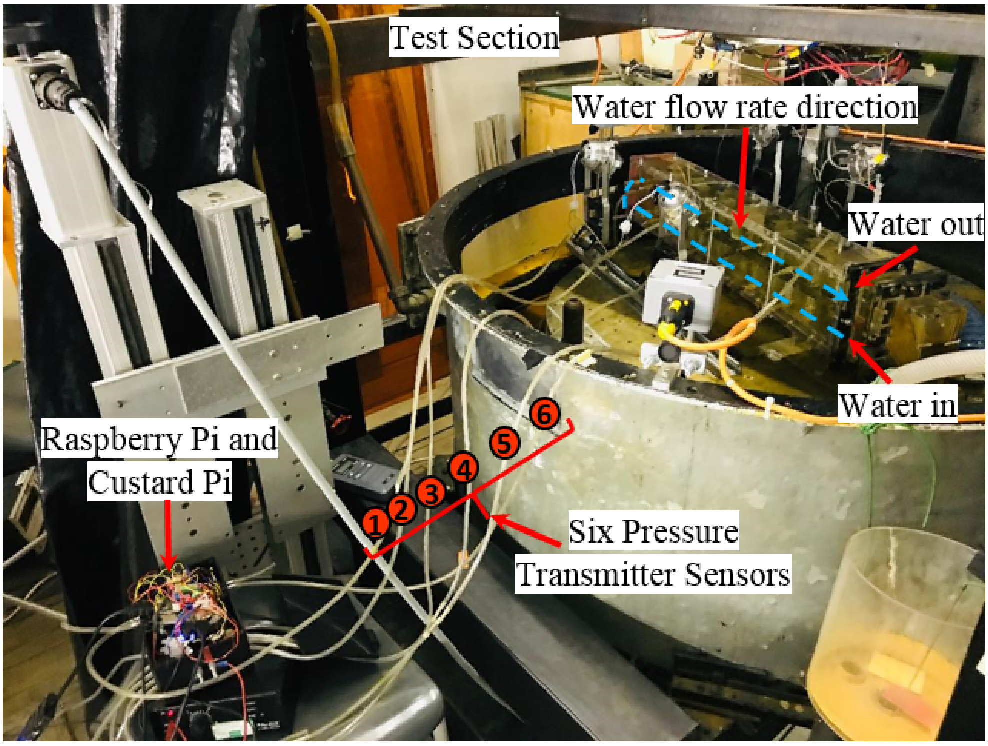


Figure 5.13 The experimental setup.

Table 5.1 Parameters used in this study.

Parameter	Value
Type of IP	IPv4
Header Size + Payload	27 bytes
Type of MQTT-QoS used	2
RasPi B power	5-volt, 700 mA
RasPi to access point link	Wireless
Monitoring server to access point link	Wireless
Monitoring server operating system	Windows 8.1
RasPi B operating System	Raspbian

5.4.1 Implementation Using Python

After the hardware was connected and seen to be working correctly, the Python script file was constructed on the RasPi and monitoring server to read and receive the sensor values, respectively. The collected data was then forwarded to the MQTT broker via the internet. On the other hand, the monitoring server received and saved the transmitted data into the database table. In order to implement the proposed system, there are several requirements to be met:

5.4.1.1 Prerequisites

The inherent requirements of the proposed protocol are listed below:

- **Python 2.7:** is installed in both devices; RasPi and the monitoring server which helps to read the sensor values, process and forward the data to the monitoring server. It also assisted in receiving, plotting and storing these data sets on the monitoring server database.
- **SQLite3 database:** is used to save the data in the monitoring server database. SQLite3 is integrated with Python using a Sqlite3 module. Thus, it did not require any further installation.
- **Python pip:** is a package management system which was used to install and manage software packages that were written in Python.
- **Paho MQTT Client:** is a lightweight publish and subscribe system that allows publishing and receiving messages as a client. It was installed in both RasPi and the monitoring server.
- **MQTT Broker:** helps to establish the connection and exchange information between the publisher and subscriber devices. The open service MQTT broker (iot.eclipse.org) was used in this study.

- **Python matplotlib:** is a Python library that draws graphs and plots the data during real-time communication. It was installed on the monitoring server.

5.4.2 Monitoring Data

Matplotlib is a plotting library for the Python programming language and was used to plot the collected data during the real-time communication on the monitoring server. The Python script ran for twenty-five minutes on the monitoring server to plot figures in real-time.

Figures 5.14 to 5.19 show the graphic display of the data received from the six sensors. Each figure presents the data from a particular sensor, and each has three panels. In each case, the uppermost panel, labelled (a), presents the actual data gathered by the sensor. Because the sensors were deployed in different locations in the test section, each sensor measured a slightly different pressure than the other sensors. The middle panel, labelled (b), presents the data published, transmitted from the RasPi to the monitoring server based on EEDF policy. The EEDF protocol eliminated the continuous reading of similar data and drops these data before published it to the MQTT broker. The bottom panel, labelled (c), shows typical data received from the MQTT broker and retrieved to the original data collected, based on the PNDD algorithm.

To assess how accurately, reliably and economically the PNDD algorithm reproduced the original date we compare, for each sensor, panels (a) and (c) in Figures 5.14 to 5.19.

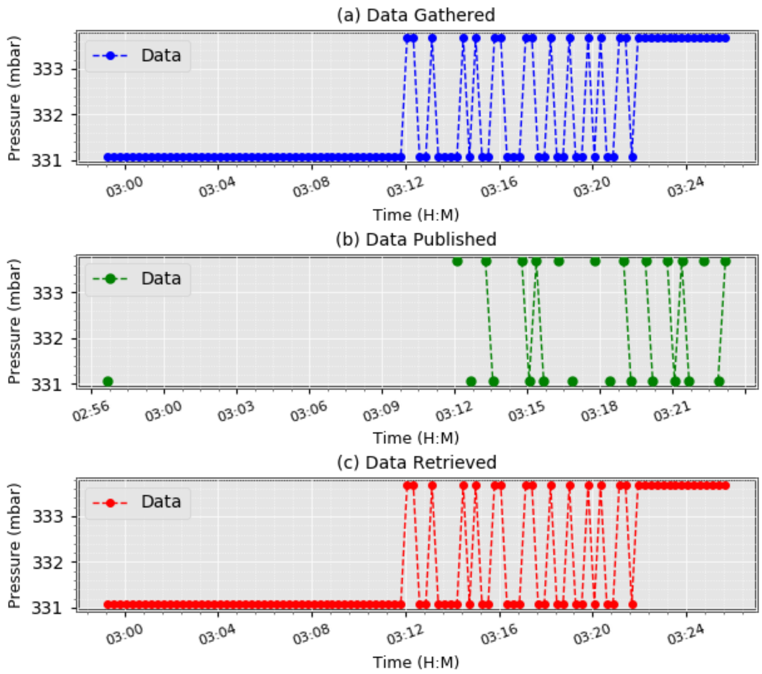


Figure 5.14 Data from Sensor 1.

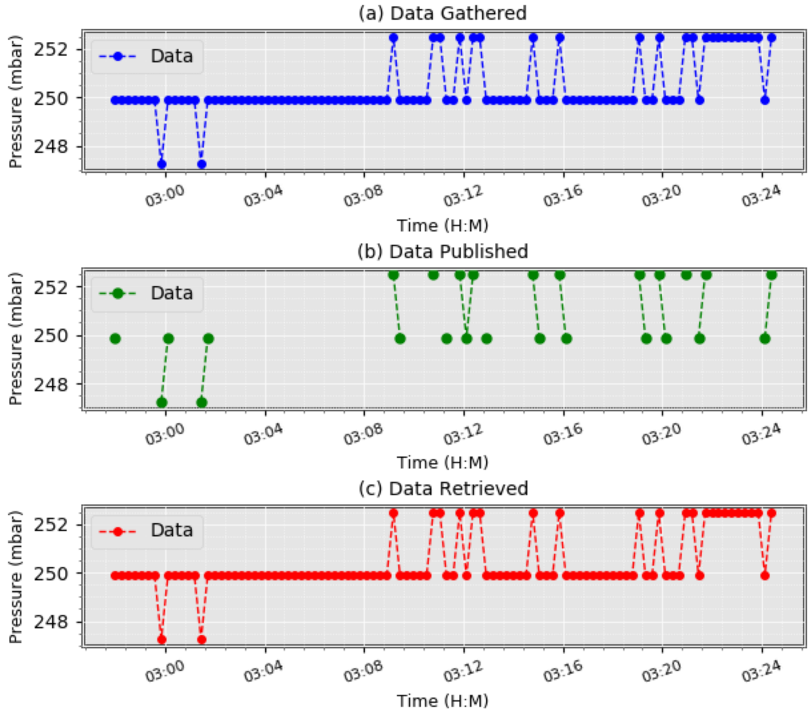


Figure 5.15 Data from Sensor 2.

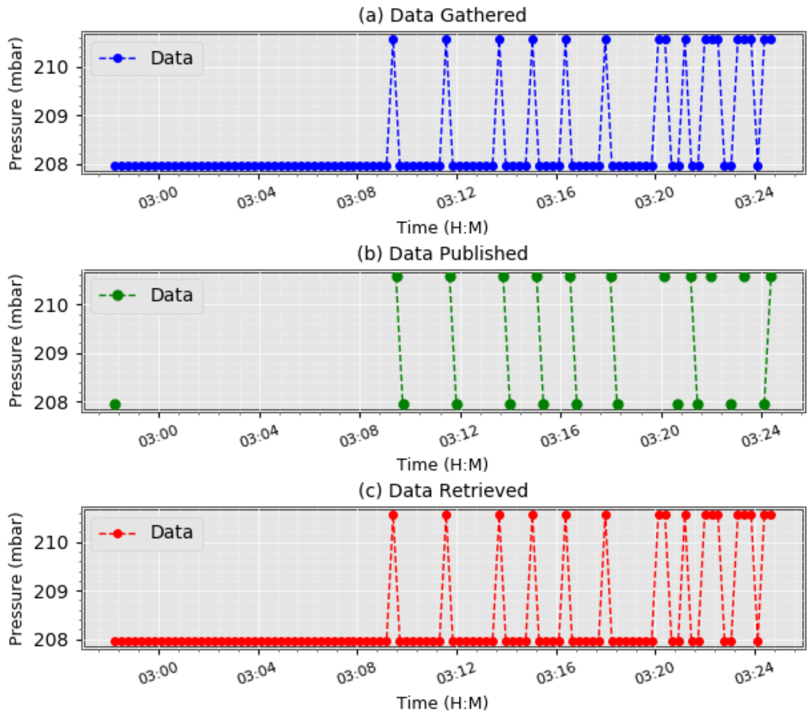


Figure 5.16 Data from Sensor 3.

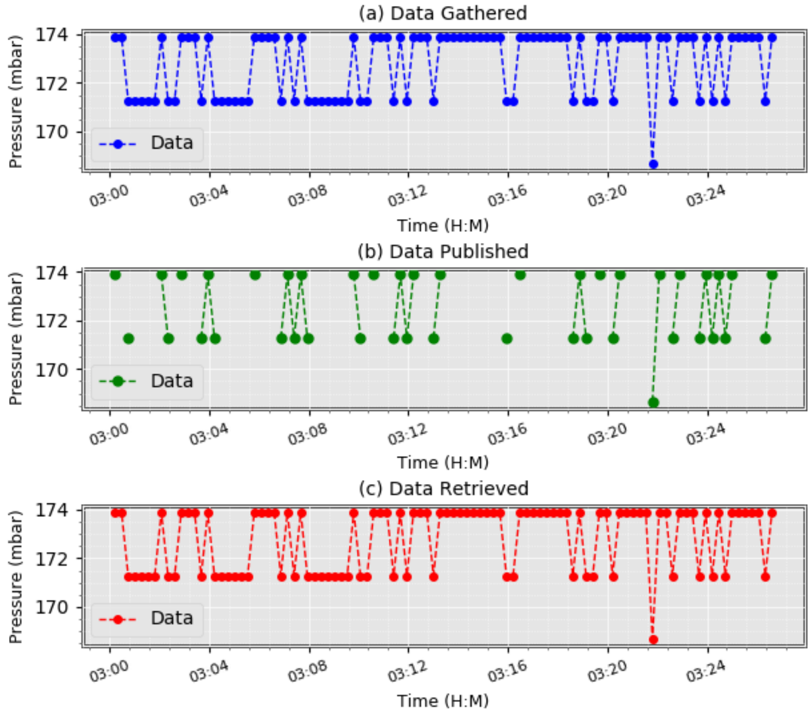


Figure 5.17 Data from Sensor 4.

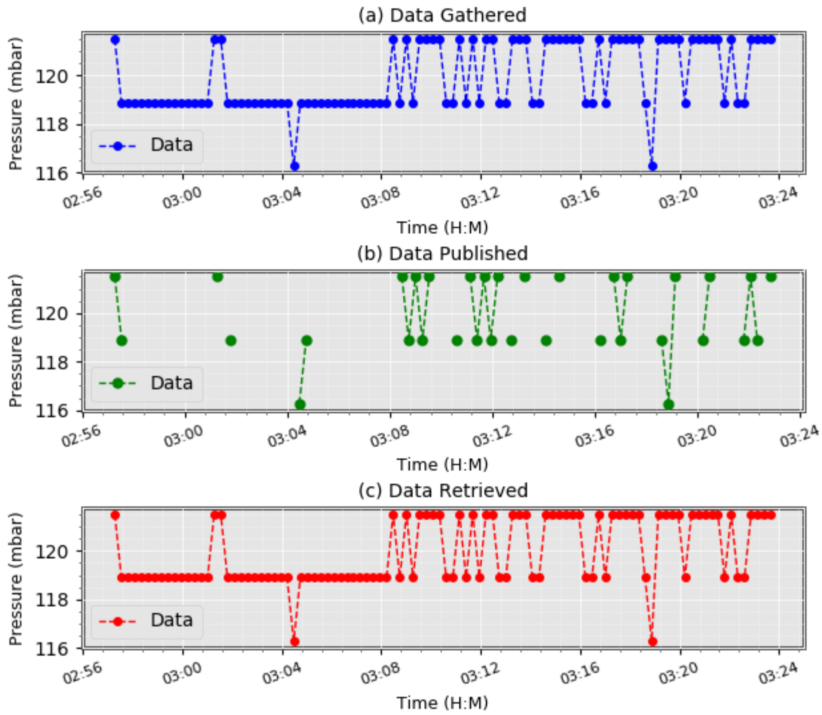


Figure 5.18 Data from Sensor 5.

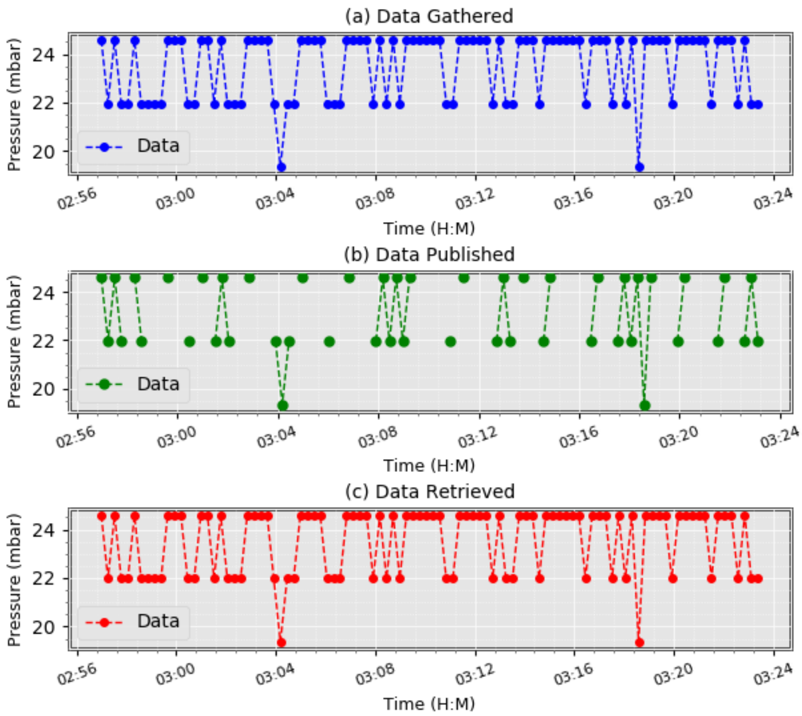


Figure 5.19 Data from Sensor 6.

5.4.3 Experimental Results

In this sub-section, we present the experimental results obtained by using the proposed algorithms. This section is divided into two sub-sections. In the first, we explain the outcomes of the proposed scheme for IoT device, and in the second, we introduce the outcomes for a traditional network.

5.4.3.1 IoT Device Results

It is clear from the Figures 5.14 to 5.19 that use of the energy-efficient data forward (EEDF) and the predicted non-dispatch data (PNDD) algorithms reduced data redundancy which reduced the number of data transmissions. This had a positive impact on the overall energy consumption of the publisher device, (RasPi) as shown in Figure 5.20. It is obvious from this figure, when the proposed EEDF algorithm is implemented, that the energy consumed by each sensor is far less than without algorithm. However, with the algorithm in use we see that each sensor consumes a different amount energy. This is because of the data redundancy is different for each node. It is also shown, without the algorithm that not only was the energy usage higher but it was higher for every sensor node. This was because the RasPi sent all the data gathered without any filtering.

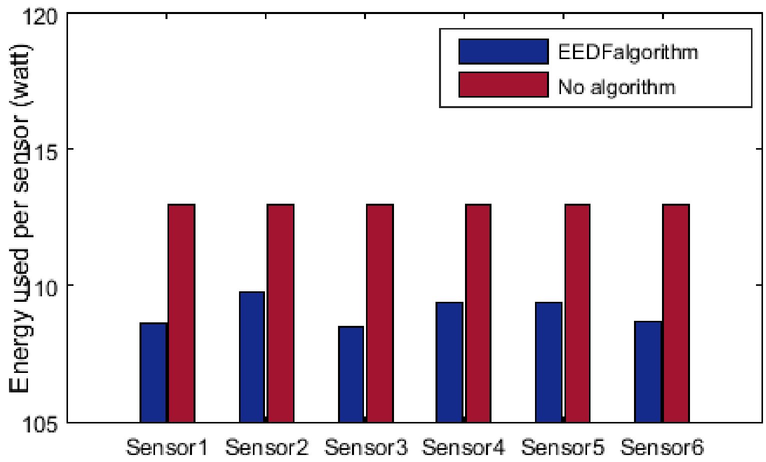


Figure 5.20 Total energy consumed by each sensor with and without the proposed algorithm.

The total network energy is defined as the summation of energy used by each sensor for each transmission, see Figure 5.21. This figure presents a comparison between using the EEDF algorithm and no-algorithm scenarios in terms of total energy consumption. We see a 3.5% reduction in total energy consumption with the addition of the proposed algorithm which would lead to prolong network lifetime.

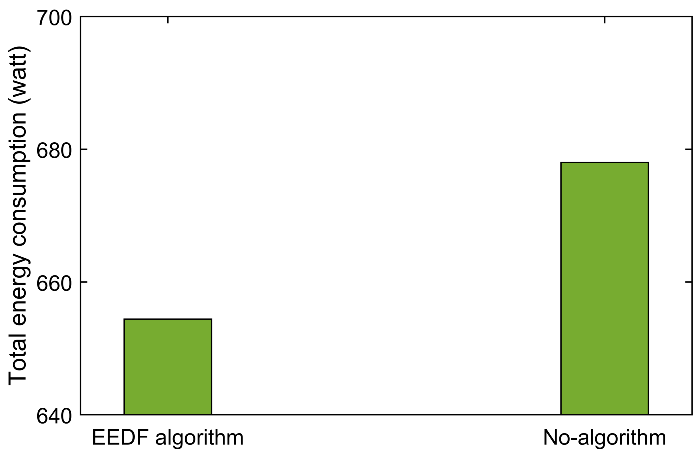


Figure 5.21 The overall energy consumption in the system with and without the proposed algorithm.

In Figure 5.22, the no-algorithm scenario, the transmitted packets by RasPi require 27 bytes. This includes the IPv4 header which is 20 bytes plus the payload of each sensor (7 bytes). By calculating the overall value for six sensors for 25 minutes, the RasPi transmitted 16,200 bytes of data. However, when the proposed algorithm was implemented the number of bytes sent by the RasPi could be calculated based on the redundant data gathered by each sensor that would be removed by the proposed algorithm. The proposed technique significantly reduces the total data transmitted by up to 76%, and correspondingly reduces the volume of data traffic on the network.

Furthermore, Figure 5.23 shows that the proposed scheme also reduces data storage in both the cloud and monitoring server. It is seen from this figure that the data storage is reduced by 80% when the proposed new forms of the EEDF and PNDD algorithms are implemented.

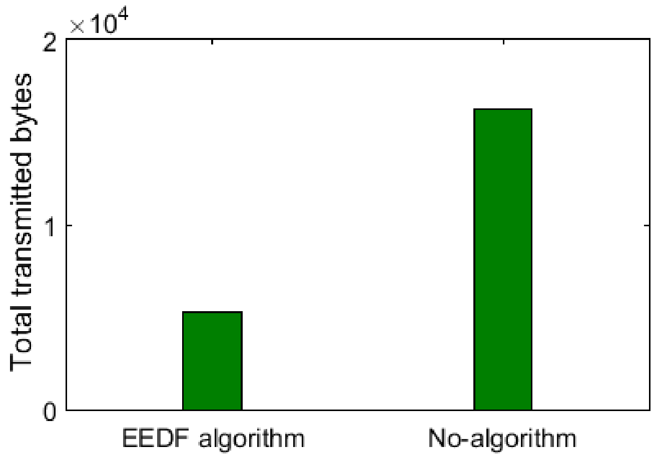


Figure 5.22 Total bytes transmitted by RasPi with and without the proposed algorithms.

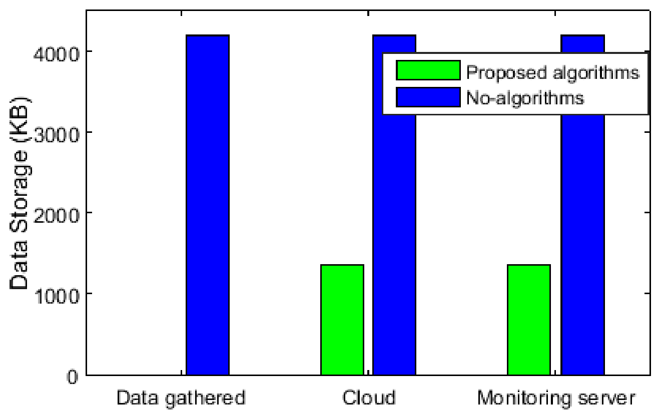


Figure 5.23 Data storage with and without the proposed algorithm.

5.4.3.2 Traditional Network Results

The literature review revealed that IoT devices substantially increase the volume of data transmitted over the internet [22]. These data are routed via the existing traditional network infrastructure, and it is important to minimise network traffic load, bandwidth and storage on the traditional networks to meet the requirements for the expected growth in IoT devices. This study proposes reducing data transmissions which could lead to significantly reducing the processing time and bandwidth used in the overall network (i.e., from publisher to subscriber device via MQTT broker). In this chapter, the traceroute command is applied in the RasPi and monitoring server using traceroute and tracert commands, respectively (see Figures 5.24 and 5.25). This command is used to determine the number of intermediate routers which are involved in the transmission process from the source to the destination.

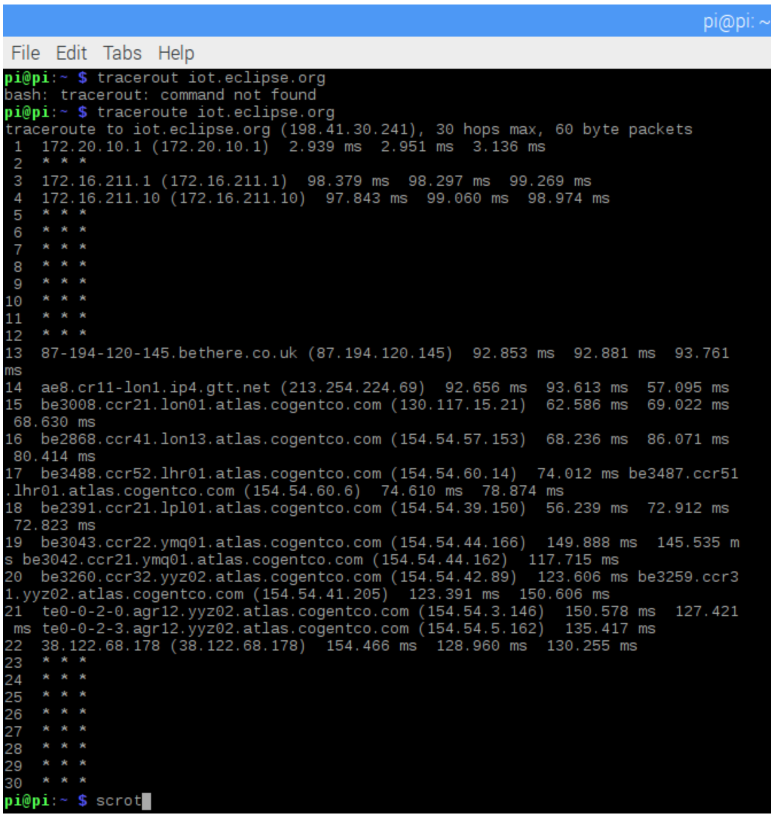


Figure 5.24 Traceroute command in Linux.

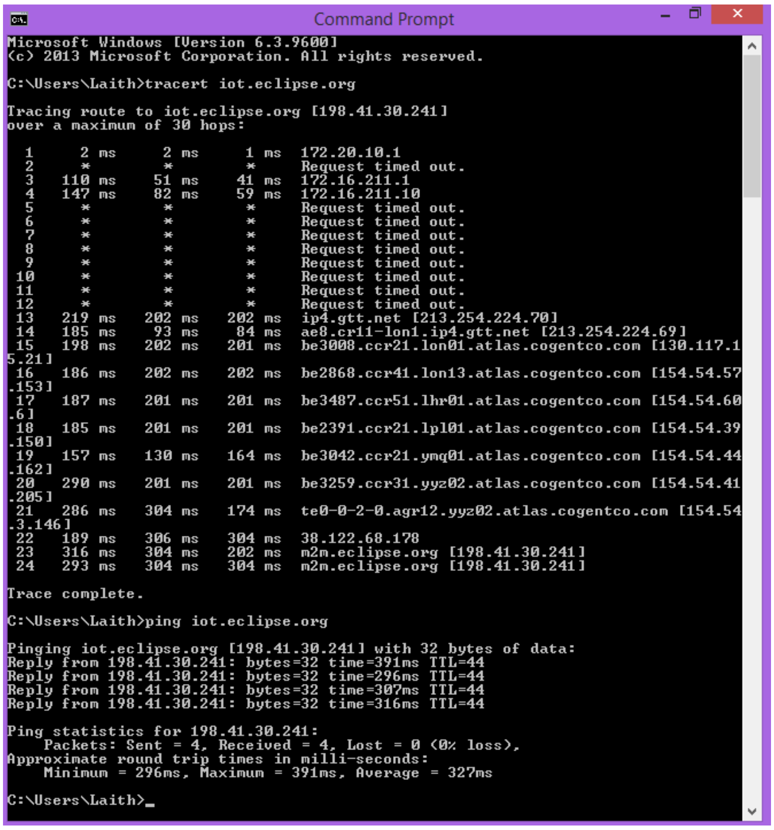


Figure 5.25 Traceroute command in Windows.

Figure 5.26 shows the difference between the total number of transmitted bytes with and without implementing the proposed algorithms. The calculation of the number of devices was achieved in two stages: the first stage was from the RasPi to the MQTT broker, and the second stage was from the monitoring server to the MQTT broker, and was based on Figures 5.24 and 5.25. The summation of both stages gives the total number of devices which took part in the communication between the RasPi and the monitoring server. The total transmitted bytes was calculated as $the\ total\ number\ of\ devices \times total\ number\ of\ transmitted\ bytes\ of\ each\ sensor$. Figure 5.26 shows that the total number of transmitted bytes falls by approximately two-third with the implementation of the proposed EEDF and PNDD algorithms. This would be a means that using the proposed development would substantially reduce the amount of data, bandwidth used, processing time in each device on the transmission path over the entire network.

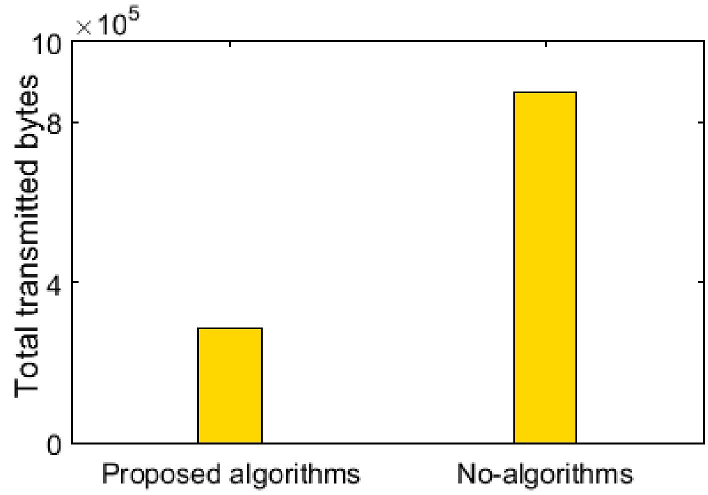


Figure 5.26 Total number of transmitted bytes from the RasPi to the monitoring server with and without the proposed algorithms.

5.5 Summary

Energy-saving has become one of the most pressing needs of WSNs and loT networks. Redundant data increases the unnecessary/unwanted processing and transmission of data which consumes energy and reduces network lifetime. Here a proposed method for more efficient data transmission has been presented for IoT-based real-time monitoring and communication networks. The proposed method removes redundant data transmission and reduces the communication costs related to sending unnecessary data.

The test system was divided into two phases: *Phase* 1 involves six sensors to gather the data from the test section and forward it to the RasPi B. The RasPi filtered and performed data redundancy checks before sending the data to the end-user via the MQTT broker. It only sends data acquired from the sensors after removing redundant data. *Phase* 2 listens to the data that comes from the RasPi via the MQTT broker and predicts the data which was eliminated by the RasPi. Simulation results confirmed that the proposed scheme outperformed the no-algorithm protocol, minimising the energy consumption, database storage, and total bytes transmitted by approximately 3.5%, 80% and 76%, respectively. The proposed system also provided the end-user with a remote monitoring system that can periodically check and track the performance of the sensors/IoT device based on published and subscribed data.

The EEDF and PNDD protocols developed here have been successfully demonstrated using RasPi B device in a controlled environment. It is judged that the proposed technique would be valuable if implemented in a remote monitoring scenario for mission-critical applications with a huge number of battery-powered nodes with limited network connectivity.

Chapter 6

Conclusions and Suggested Future Work

This thesis has presented a state-of-the-art technological development for wireless sensor networks (WSNs) that, if widely implemented, could greatly enhance the Internet of Things (IoT) in terms of reducing energy consumption of sensor nodes and thus extending network lifetimes. In this chapter, the contribution to the knowledge of the research is highlighted and potential future directions presented.

6.1 Conclusions

WIRELESS sensor networks (WSNs) are a key enabling technology for the internet of things (IoT). Sensor nodes capable of detecting the required information, performing some processing and communicating with other connected nodes are the main component of these networks. However, the life of these nodes is often restricted by being powered by a battery with a limited life, constraining processing ability, memory, and radio communications. Energy efficiency is one of the most crucial issues for WSNs; it is not rational to waste energy on protocol overheads, the transmission of unneeded data or non-optimised transmission of data packets, especially retransmissions, due to inefficient scheduling and routing algorithms. The main goal of this thesis has been to identify and introduce measures to correct the corresponding energy wastages in order to prolong the network lifetime.

This thesis began with the question of which strategies have been used for WSN-assisted IoT applications in order to find a suitable approach that would reduce energy drainage from the sensor nodes and improve network performance.

The major contributions of this thesis are summarised as:

- It is shown that it is important to balance the load traffic between the nodes on the forward path. A node that receives and forwards data using fewer forwarding nodes consumes less energy than a node which receives and forwards data using a greater number of forwarding nodes.

- Data packets that come from further away require access to a greater number of intermediate nodes, and thus consume disproportionately more energy. We show that giving data packets from the farthest nodes higher priority over data packets from nearby nodes will reduce retransmissions and save this energy.

- Increasing the transmission distances between a node and its neighbour nodes, and the cluster head (CH) nodes and the base station (BS) increases the energy consumption of these nodes. This is because the power required for transmission increases rapidly with the distance between nodes.

- We show that interference increases with an increase in the number of neighbouring nodes and that can cause an increase in the number of retransmissions. We propose an algorithm for avoiding higher node interference to help minimise the overall energy consumption of the network.

- We also confirm that an increase in the number of neighbour nodes around a node has a negative impact on the network lifetime of WSNs due to overhearing. Overhearing can be a significant problem when node density is high and the traffic load is heavy. The algorithm we propose for avoiding higher node interference will also reduce overhearing.

- Redundant data increases energy consumption during data processing and transmitting. Therefore, the energy consumption can be minimised by avoiding unnecessary operations of a node.
- A remote monitoring system for the end-user that checks and tracks the performance of sensors/IoT devices during real-time communication could reduce production and maintenance costs and improve the entire IoT system.

6.2 Future Work

It is considered there were some limitations to the work carried out and reported here that could be improved. Following are possible future directions that could be considered to improve these research findings:

- This study has proposed a scheduling algorithm that assigns a high priority for packets that come from greater distances and access a higher number of nodes, these should be the first to be routed to the ultimate receiver in order to conserve energy. However, some sensor nodes need to deliver their packets within a certain time. Thus, research needs to be undertaken to improve the performance of the proposed algorithm by including the mission-critical of IoT sensors in the scheduling classification in order to improve the quality of service system.
- In this research, the proposed algorithms fall under homogeneous WSNs. However, IoT-based WSNs include a variety of sensors and networks that can connect and communicate with each other. The proposed solutions proposed for homogeneous networks would not be suitable for heterogeneous networks. Therefore, the concept of heterogeneous networks needs to be considered.
- Using mobile nodes to collect the data from the sensing field was not considered in this research. The BS could be a mobile node moving in the sensing field to collect data from sensor nodes/CH nodes. The routing and clustering strategy followed for a fixed BS should be redesigned to suit a mobile BS.
- One of the main issues of IoT applications is to develop a security-aware packet scheduling algorithm that can effectively and efficiently improve security while dispatching the data through the internet links.

References

[1] Shancang Li, Li Da Xu, and Shanshan Zhao. The Internet of Things: a survey. *Information Systems Frontiers*, 17(2):243–259, Apr. 2015.

[2] William Stallings. *Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud*. Addison-Wesley Professional, 2015.

[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys Tutorials*, 17(4):2347–2376, Jun. 2015.

[4] V. Kumar, S. Mohan, and R. Kumar. A Voice Based One Step Solution for Bulk IoT Device Onboarding. In *2019 16th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pages 1–6, Jan. 2019.

[5] P.P. Ray. A Survey on Internet of Things Architectures. *Journal of King Saud University -Computer and Information Sciences*, 30(3):291–319, Oct. 2018.

[6] Simon Elias Bibri and John Krogstie. Smart Sustainable Cities of the Future: An Extensive Interdisciplinary Literature Review. *Sustainable Cities and Society*, 31:183 – 212, May 2017.

[7] Kemal Akkaya and Mohamed Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325–349, May 2005.

[8] Laith Farhan and Rupak Kharel. *Internet of Things: Vision, Future Directions and Opportunities*, volume 29. Springer International Publishing, Cham, 2019.

[9] J. So and H. Byun. Load-Balanced Opportunistic Routing for Duty-Cycled Wireless Sensor Networks. *IEEE Transactions on Mobile Computing*, 16(7):1940–1955, Jul. 2017.

[10] Murad Khan, Muhammad Babar, Syed Hassan Ahmed, Sayed Chhattan Shah, and Kijun Han. Smart city designing and planning based on big data analytics. *Sustainable Cities and Society*, 35:271–279, Nov. 2017.

[11] Awais Ahmad, Mazhar Rathore, Muhammad, Anand Paul, and Bo-Wei Chen. Data Transmission Scheme Using Mobile Sink in Static Wireless Sensor Network. *Journal of Sensors*, 2015:8 pages, Jun. 2015.

[12] Jasenka Dizdareviundefined, Francisco Carpio, Admela Jukan, and Xavi Masip-Bruin. A Survey of Communication Protocols for Internet of Things and Related Challenges of Fog and Cloud Computing Integration. *ACM Computing Surveys*, 51(6), Jan. 2019.

[13] Shabna VC, Jamshid K, and S. Manoj Kumar. Energy minimization by removing data redundancy in wireless sensor networks. In *2014 International Conference on Communication and Signal Processing*, pages 1658–1663, Apr. 2014.

[14] Laith Farhan, Rupak Kharel, Omprakash Kaiwartya, Mohammad Hammoudeh, and Bamidele Adebisi. Towards Green Computing for Internet of Things: Energy Oriented Path and Message Scheduling Approach. *Sustainable Cities and Society*, 38:195–204, Apr. 2018.

[15] Laith Farhan, Omprakash Kaiwartya, Laith Alzubaidi, Waled Gheth, Eric Dimla, and Rupak Kharel. Toward Interference Aware IoT Framework: Energy and Geo-Location-Based-Modeling. *IEEE Access*, 7:56617–56630, Apr. 2019.

[16] Barry M Leiner, Vinton G Cerf, David D Clark, Robert E Kahn, Leonard Kleinrock, Daniel C Lynch, Jon Postel, Larry G Roberts, and Stephen Wolff. A brief history of the Internet. *ACM SIGCOMM Computer Communication Review*, 39(5):22–31, Oct. 2009.

[17] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. Context Aware Computing for the Internet of Things: A Survey. *IEEE Communications Surveys Tutorials*, 16(1):414–454, May 2014.

[18] Brian Engard. An Internet History Timeline: From the 1960s to Now, 2016 (accessed February 28, 2019). https://online.jefferson.edu/communications/internet-history-timeline.

[19] Melanie Swan. Sensor Mania! The Internet of Things, Wearable Computing, Objective Metrics, and the Quantified Self 2.0. *Journal of Sensor and Actuator Networks*, 1:217–253, Nov. 2012.

[20] Mehdi Imani, Abolfazl Qiasi Moghadam, Nasrin Zarif, Maaruf Ali, Omekolsoom Noshiri, Kimia Faramarzi, Hamid Arabnia, and Majid Joudaki. A Comprehensive Survey on Addressing Methods in the Internet of Things, Apr. 2018.

[21] Daniel Hoornweg and Kevin Pope. Population predictions for the world's largest cities in the 21st century. *Environment and Urbanization*, 29(1):195–216, Sept. 2017.

[22] C. Beecks, A. Grass, and S. Devasya. Metric Indexing for Efficient Data Access in the Internet of Things. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 5132–5136, Dec. 2018.

[23] F. Ramparany and Q. H. Cao. A semantic approach to IoT data aggregation and interpretation applied to home automation. In *2016 International Conference on Internet of Things and Applications (IOTA)*, pages 23–28, Jan. 2016.

[24] Y. Zhang, J. Ren, J. Liu, C. Xu, H. Guo, and Y. Liu. A Survey on Emerging Computing Paradigms for Big Data. *Chinese Journal of Electronics*, 26(1):1–12, Jan. 2017.

[25] Reinsel David, Gantz John, and Rydning John. Data Age 2025: The Evolution of Data to Life-Critical . https://assets.ey.com/content/dam/ey-sites/eycom/en_gl/topics/workforce/Seagate-WP-DataAge2025-March-2017.pdf.

[26] Abdelrahman Abuarqoub, Mohammad Hammoudeh, Bamidele Adebisi, Sohail Jabbar, Ahcne Bounceur, and Hashem Al-Bashar. Dynamic Clustering and Management of Mobile Wireless Sensor Networks. *Comput. Netw.*, 117(C):62–75, Apr. 2017.

[27] S. Mubeen, S. A. Asadollah, A. V. Papadopoulos, M. Ashjaei, H. Pei-Breivold, and M. Behnam. Management of Service Level Agreements for Cloud Services in IoT: A Systematic Mapping Study. *IEEE Access*, 6:30184–30207, Aug. 2018.

[28] S. Narula, A. Jain, and Prachi. Cloud Computing Security: Amazon Web Service. In *2015 Fifth International Conference on Advanced Computing Communication Technologies*, pages 501–505, Feb. 2015.

[29] S. S. Mishra and A. Rasool. IoT Health care Monitoring and Tracking: A Survey. In *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 1052–1057, Apr. 2019.

[30] M. Amarlingam, P. K. Mishra, K. V. V. D. Prasad, and P. Rajalakshmi. Compressed sensing for different sensors: A real scenario for WSN and IoT. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 289–294, Dec. 2016.

[31] Ali Ahmad, Ming Yu, Chakraborty Sagnik, and Iram Saima. A Comprehensive Survey on Real-Time Applications of WSN. *Future Internet*, 9(4), Nov. 2017.

[32] P. Thota and Y. Kim. Implementation and Comparison of M2M Protocols for Internet of Things. In *2016 4th Intl Conference on Applied Computing and Information Technology*, Dec. 2016.

[33] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The Internet of Things: A survey. *Computer Networks*, 54(15):2787–2805, Oct. 2010.

[34] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645–1660, Sept. 2013.

[35] A. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, and Q. Z. Sheng. IoT Middleware: A Survey on Issues and Enabling Technologies. *IEEE Internet of Things Journal*, 4(1):1–20, Feb. 2017.

[36] O. B. Sezer, E. Dogdu, and A. M. Ozbayoglu. Context-Aware Computing, Learning, and Big Data in Internet of Things: A Survey. *IEEE Internet of Things Journal*, 5(1):1–27, Feb. 2018.

[37] Arroyo Patricia, Herrero Jose, Luis, Suarez Jose, Ignacio, and Lozano Jesus. Wireless Sensor Network Combined with Cloud Computing for Air Quality Monitoring. *Sensors*, 19:691, Feb. 2019.

[38] V. C. Emeakaroha, N. Cafferkey, P. Healy, and J. P. Morrison. A Cloud-Based IoT Data Gathering and Processing Platform. In *2015 3rd International Conference on Future Internet of Things and Cloud*, pages 50–57, Aug. 2015.

[39] Partha Pratim Ray. A survey of IoT cloud platforms. *Future Computing and Informatics Journal*, 1(1):35–46, 2016.

[40] D. Airehrour, J. Gutierrez, and S. K. Ray. A Lightweight Trust Design for IoT Routing. In *2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computinence and Technology*, pages 552–557, Aug. 2016.

[41] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao. A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications. *IEEE Internet of Things Journal*, 4(5):1125–1142, Oct. 2017.

[42] Bragg Graeme, McLachlan. *Standards-Based Internet of Things Sub-GHz Environmental Sensor Networks*. PhD thesis, University of Southampton, 2017.

[43] Silvia Hagen. IPV6 Essentials. *Grundlagen - Funktionalität, Integration 2nd edition Sunny Edition*, 2006.

[44] X. Wang, H. Cheng, and Y. Yao. Addressing-Based Routing Optimization for 6LoWPAN WSN in Vehicular Scenario. *IEEE Sensors Journal*, 16(10):3939–3947, May 2016.

[45] O. Amale and R. Patil. IoT Based Rainfall Monitoring System Using WSN Enabled Architecture. In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pages 789–791, Mar. 2019.

[46] A. M. Brohi, Y. A. Malkani, and M. S. Chandio. Data provenance in Wireless Sensor Networks (WSNs): A review. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pages 1–5, Mar. 2018.

[47] J. Shen, A. Wang, C. Wang, P. C. K. Hung, and C. F. Lai. An Efficient Centroid-Based Routing Protocol for Energy Management in WSN-Assisted IoT. *IEEE Access*, 5:18469–18479, Sept. 2017.

[48] H. Ding, D. Zhao, Y. Zhao, and Z. Liu. Application of WSN of Polling System with Vacations and Using Gated Services in Forest Fire Prevention. In *2010 International Conference on Internet Technology and Applications*, pages 1–4, Aug. 2010.

[49] S. Nagaraj and R. V. Biradar. Applications of wireless sensor networks in the real-time ambient air pollution monitoring and air quality in metropolitan cities : A survey. In *2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, pages 1393–1398, Aug. 2017.

[50] A. Bhavsar and H. Arolkar. ZigBee based Network Architecture for Animal Health Monitoring. In *2015 1st International Conference on Next Generation Computing Technologies (NGCT)*, pages 398–402, Sept. 2015.

[51] Buratti Chiara, Conti Andrea, Dardari Davide, and Verdone Roberto. An Overview on Wireless Sensor Networks Technology and Evolution. *Sensors*, 9:6869–6896, Aug. 2009.

[52] L. Herrera, A. Calveras, and M. Catalan. A two-way radio communication across a multi-hop wireless sensor network based on a commercial IEEE 802.15.4 compliant platform. *Procedia Engineering*, 25:1045–1048, Dec. 2011.

[53] Asim Zeb, A. K. M. Muzahidul Islam, Mahdi Zareei, Ishtiak Al Mamoon, Nafees Mansoor, Sabariah Baharun, Yoshiaki Katayama, and Shozo Komaki. Clustering Analysis in Wireless Sensor Networks: The Ambit of Performance Metrics and Schemes Taxonomy. *International Journal of Distributed Sensor Networks*, 12(7):4979142, Jul. 2016.

[54] Jau-Yang Chang and Ting-Huan Shen. An Efficient Tree-Based Power Saving Scheme for Wireless Sensor Networks With Mobile Sink. *IEEE Sensors*, 16(20):7545–7557, Oct. 2016.

[55] Sarwesh P, N. Shekar V. Shet, and Chandrasekaran K. Energy efficient network architecture for IoT applications. In *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Oct. 2015.

[56] Puneet Azad and Vidushi Sharma. Cluster Head Selection in Wireless Sensor Networks under Fuzzy Environment. *International Scholarly Research Notices*, 2013:8 pages, Feb. 2013.

[57] K. Laubhan, K. Talaat, S. Riehl, T. Morelli, A. Abdelgawad, and K. Yelamarthi. A four-layer wireless sensor network framework for IoT applications. In *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1–4, Oct. 2016.

[58] D Saha, M R Yousuf, and M A Matin. Energy Efficient Scheduling Algorithm For S-MAC Protocol in Wireless Sensor Network. *International Journal of Wireless and Mobile Networks*, 3(6):129–140, Dec. 2011.

[59] T. M. Behera, S. K. Mohapatra, U. C. Samal, M. S. Khan, M. Daneshmand, and A. H. Gandomi. Residual Energy-Based Cluster-Head Selection in WSNs for IoT Application. *IEEE Internet of Things Journal*, 6(3):5132–5139, Jun. 2019.

[60] I. S. H. Martinez, I. P. O. J. Salcedo, and I. B. S. R. Daza. IoT application of WSN on 5G infrastructure. In *2017 International Symposium on Networks, Computers and Communications (ISNCC)*, pages 1–6, May 2017.

[61] João H Kleinschmidt. Genetic Algorithms for Wireless Sensor Networks. In *Encyclopedia of Artificial Intelligence*, pages 755–758. IGI Global, Jan. 2009.

[62] S. Khriji, R. Cheour, M. Goetz, D. El Houssaini, I. Kammoun, and O. Kanoun. Measuring energy consumption of a wireless sensor node during transmission: panstamp. In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, pages 274–280, May 2018.

[63] Pascale Minet. *Energy Efficient Routing*. Bentham Science, 2009.

[64] A. H. Ab.Halim and K. Zen. MAC protocol to reduce packet collision in wireless sensor network. In *2008 International Conference on Computer and Communication Engineering*, pages 1290–1293, May 2008.

[65] Ye Chen, Wei Liu, Tian Wang, Qingyong Deng, Anfeng Liu, and Houbing Song. An adaptive retransmit mechanism for delay differentiated services in industrial WSNs. *EURASIP Journal on Wireless Communications and Networking*, 2019(1):258, 2019.

[66] D. Agarwal, R. R. Rout, and S. Ravichandra. Detection of node-misbehavior using overhearing and autonomous agents in wireless Ad-Hoc networks. In *2015 Applications and Innovations in Mobile Computing (AIMoC)*, pages 152–157, Feb. 2015.

[67] Amitangshu Pal and Asis Nasipuri. Distributed Routing and Channel Selection for Multi-Channel Wireless Sensor Networks. *Sensor and Actuator Networks*, 6:10, Jul. 2017.

[68] C. J. Liyana Arachchige, S. Venkatesan, and N. Mittal. An Asynchronous Neighbor Discovery Algorithm for Cognitive Radio Networks. In *2008 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pages 1–5, Oct. 2008.

[69] R. Margolies, G. Grebla, T. Chen, D. Rubenstein, and G. Zussman. Panda: Neighbor Discovery on a Power Harvesting Budget. *IEEE Journal on Selected Areas in Communications*, 34(12):3606–3619, Dec. 2016.

[70] Nadeem Javaid, Abdul Majid, Arshad Sher, Wazir Zada Khan, and Mohammed Y. Aalsalem. Avoiding Void Holes and Collisions with Reliable and Interference-Aware Routing in Underwater WSNs. *Sensors*, 18(9):1–25, Sept. 2018.

[71] Musaloiu-E Razvan and Terzis Andreas. Minimising the effect of WiFi interference in 802.15.4 wireless sensor networks. *International Journal Sensor Networks*, 3(1):43–54, Dec. 2008.

[72] Ravneet Kaur, Deepika Sharma, and Navdeep Kaur. Comparative Analysis of Leach And Its Descendant Protocols In Wireless Sensor Network. *International Journal of P2P Network Trends and Technology (IJPTT)*, 3:22–27, Jan. 2013.

[73] Xuesong Liu and Jie Wu. A Method for Energy Balance and Data Transmission Optimal Routing in Wireless Sensor Networks. *Sensors*, 19(13):1–25, Jul. 2019.

[74] J. Zhang, Q. Wu, F. Ren, T. He, and C. Lin. Effective Data Aggregation Supported by Dynamic Routing in Wireless Sensor Networks. In *2010 IEEE International Conference on Communications*, pages 1–6, May 2010.

[75] S. Lindsey and C. S. Raghavendra. PEGASIS: Power-efficient gathering in sensor information systems. In *Proceedings, IEEE Aerospace Conference*, Mar. 2002.

[76] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. An Application-Specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Transactions on Wireless Communications*, 1(4):660–670, Oct. 2002.

[77] J. Liang, J. Wang, and J. Chen. A Delay-Constrained and Maximum Lifetime Data Gathering Algorithm for Wireless Sensor Networks. In *2009 Fifth International Conference on Mobile Ad-hoc and Sensor Networks*, pages 148–155, Dec. 2009.

[78] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados. Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey. *IEEE Communications Surveys Tutorials*, 15(2):551–591, Jul. 2013.

[79] J. Zhang, P. Hu, F. Xie, J. Long, and A. He. An Energy Efficient and Reliable In-Network Data Aggregation Scheme for WSN. *IEEE Access*, 6:71857–71870, Nov. 2018.

[80] E. I. Oyman and C. Ersoy. Multiple Sink Network Design Problem in Large Scale Wireless Sensor networks. In *IEEE International Conference on Communications*, pages 3663–3667, Jun. 2004.

[81] Ossama Younis and Sonia Fahmy. HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on mobile computing*, 3(4):366–379, 2004.

[82] S. Lindsey, C. Raghavendra, and K. M. Sivalingam. Data gathering algorithms in sensor networks using energy metrics. *IEEE Transactions on Parallel and Distributed Systems*, 13(9):924–935, Sep 2002.

[83] Wendi B. Heinzelman. *Application- Specific Protocol Architectures for Wireless Networks*. PhD thesis, MIT, 2000.

[84] K. Singh. WSN LEACH based protocols: A structural analysis. In *2015 International Conference and Workshop on Computing and Communication (IEMCON)*, pages 1–7, Oct. 2015.

[85] Prof. Greeshma Arya, Prof. D.S Chauhan, Ridhi Khanna, Nehal Mittal, and Amandeep. Energy Aware Multi-Hop Multi-Path Hierarchical (EAMMH) Routing Protocol for Wireless Sensor Networks. *European Journal of Scientific Research*, 88(7), 2012.

[86] K. Cengiz and T. Dag. Energy Aware Multi-Hop Routing Protocol for WSNs. *IEEE Access*, 6:2622–2633, Dec. 2018.

[87] M. Eshaftri, A. Y. Al-Dubai, I. Romdhani, and M. B. Yassien. A New Energy Efficient Cluster Based Protocol for Wireless Sensor Networks. In *2015 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 1209–1214, Sept. 2015.

[88] Liu Xuxun. A Survey on Clustering Routing Protocols in Wireless Sensor Networks. *Sensors*, 12(8):11113–11153, Aug. 2012.

[89] Anitha G., Vijayakumari V., and Shankar Thangavelu. A Comprehensive Study and Analysis of LEACH and HEED Routing Protocols for Wireless Sensor Networks - With Suggestion for Improvements. *Indonesian Journal of Electrical Engineering and Computer Science*, 9(3):778–783, Mar. 2018.

[90] H. Kour and A. K. Sharma. Performance evaluation of HEED and H-HEED protocol for realistic models in WSN. In *2015 International Conference on Computer, Communication and Control (IC4)*, pages 1–5, Sept. 2015.

[91] Z. Ullah, L. Mostarda, R. Gagliardi, D. Cacciagrano, and F. Corradini. A Comparison of HEED Based Clustering Algorithms-Introducing ER-HEED. In *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pages 339–345, Mar. 2016.

[92] J. Huang, Y. Li, and H. Yu. Design and implementation of multi-hops wireless sensor network routing protocol based on HEED. In *2017 IEEE 17th International Conference on Communication Technology (ICCT)*, pages 598–601, Oct. 2017.

[93] Parul Bansal, Poonam Kundu, and Prabhjot Kaur. Comparison of LEACH and PEGASIS hierarchical routing protocols in wireless sensor networks. *International Journal on Recent Trends in Engineering & Technology*, 11(1):139, 2014.

[94] Anuradha and Gill Inderjeet, Singh. A Brief Study of Power Efficient Hierarchical Routing Protocols in Wireless Sensor Networks. *International Journal of Computer Science and Information Technologies*, 5:6911–6915, 2014.

[95] Jau-Yang Chang and Pei-Hao Ju. An energy-saving routing architecture with a uniform clustering algorithm for wireless body sensor networks. *Future Generation Computer Systems*, 35:128 –140, Jun. 2014.

[96] Misbahul Haque, Tauseef Ahmad, and Mohd. Imran. Review of Hierarchical Routing Protocols for Wireless Sensor Networks. In Yu-Chen Hu, Shailesh Tiwari, Krishn K. Mishra, and Munesh C. Trivedi, editors, *Intelligent Communication and Computational Technologies*, pages 237–246, Singapore, 2018. Springer Singapore.

[97] W. Bechkit, M. Koudil, Y. Challal, A. Bouabdallah, B. Souici, and K. Benatchba. A new weighted shortest path tree for convergecast traffic routing in WSN. In *2012 IEEE Symposium on Computers and Communications (ISCC)*, pages 187–192, Jul. 2012.

[98] I. Banerjee, I. Roy, A. R. Choudhury, B. D. Sharma, and T. Samanta. Shortest path based geographical routing algorithm in wireless sensor network. In *2012 International Conference on Communications, Devices and Intelligent Systems (CODIS)*, pages 262–265, Dec. 2012.

[99] R. Tejas and T. H. M. Devi. Design and implementation of an efficient routing schemes to increase distribute lifetime efficiency in WSN. In *2016 IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, pages 637–640, May 2016.

[100] J. Cota-Ruiz, P. Rivas-Perea, E. Sifuentes, and R. Gonzalez-Landaeta. A Recursive Shortest Path Routing Algorithm With Application for Wireless Sensor Network Localization. *IEEE Sensors Journal*, 16(11):4631–4637, Jun. 2016.

[101] S. E. Khediri, A. Thaljaoui, A. Dallali, and A. Kachouri. Clustering algorithm in wireless sensor networks based on shortest path. In *2018 30th International Conference on Microelectronics (ICM)*, pages 335–338, Dec. 2018.

[102] M. Grossglauser and D. N. C. Tse. Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networking*, 10(4):477–486, Aug. 2002.

[103] M. H. Khodashahi, F. Tashtarian, M. H. Yaghmaee Moghaddam, and M. T. Honary. Optimal Location for Mobile Sink in Wireless Sensor Networks. In *2010 IEEE Wireless Communication and Networking Conference*, pages 1–6, Apr. 2010.

[104] K. Y. Bendigeri, J. D. Mallapur, and S. B. Kumbalavati. Recovery based fault tolerance in wireless sensor networks. In *2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, pages 122–128, Dec. 2017.

[105] R. P. Bulhoes, D. Passos, and C. V. N. Albuquerque. Collision probability estimation in wireless networks. In *2016 8th IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–6, Nov. 2016.

[106] Q. Chen, B. Zhou, and Z. Zhao. A minimum route-interference routing metric for multi-radio wireless mesh networks. In *2009 Fourth International Conference on Communications and Networking in China*, pages 1–4, Aug. 2009.

[107] H. Huang, G. Hu, F. Yu, and Z. Zhang. Energy-aware interference-sensitive geographic routing in wireless sensor networks. *IET Communications*, 5(18):2692–2702, Dec. 2011.

[108] D. Salvatore and S. H. Yang. Routing algorithm of WSN under interference environment. In *IET Conference on Wireless Sensor Systems (WSS 2012)*, pages 1–5, Jun. 2012.

[109] W. Liu, K. Huang, X. Zhou, and S. Durrani. Full-duplex backscatter interference networks based on time-hopping spread spectrum. *IEEE Transactions on Wireless Communications*, 16(7):4361–4377, Jul. 2017.

[110] Jims Marchang, Roderick Douglas, Bogdan Ghita, David Lancaster, and Benjamin Sanders. Dynamic Neighbour Aware Power-controlled MAC for Multi-hop Ad-hoc networks. *Ad Hoc Networks*, 75-76:119 –134, Jun. 2018.

[111] E. Pignaton de Freitas, T. Heimfarth, I. F. Netto, C. E. Pereira, A. M. Ferreira, F. R. Wagner, and T. Larsson. Handling Failures of Static Sensor Nodes in Wireless Sensor Network by Use of Mobile Sensors. In *2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications*, pages 127–134, Mar. 2011.

[112] Laith Farhan, Rupak Kharel, Omprakash Kaiwartya, M. Quiroz-Castellanos, U. Raza, and S. H. Teay. LQOR: Link Quality-Oriented Route Selection on Internet of Things Networks for Green Computing. In *2018 11th International Symposium on Communication Systems, Networks Digital Signal Processing (CSNDSP)*, pages 1–6, Jul. 2018.

[113] Fredrick A. Omondi, Enver Ever, Purav Shah, and Orhan Gemikonakli. Modelling wireless sensor networks for performability evaluation. In *Ad-hoc, Mobile, and Wireless Network*, pages 172–184, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[114] Saima Abdullah and Kun Yang. An Energy Efficient Message Scheduling Algorithm Considering Node Failure in IoT Environment. *Wireless Personal Communications*, 79(3):1815–1835, Dec. 2014.

[115] S. Sasikumar, S. Sudha, and R. Uma Maheswari. Recover link failure using ARTCP mechanism. In *IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012)*, pages 848–853, Mar. 2012.

[116] Tien Nguyen Xuan, Kim Semog, Rhee Jong Myung, and Park Sang Yoon. A Novel Dual Separate Paths (DSP) Algorithm Providing Fault-Tolerant Communication for Wireless Sensor Network. *Sensors*, 17:1–33, Jul. 2017.

[117] C. Titouna, H. Moumen, and A. A. Abba Ari. Cluster Head Recovery Algorithm for Wireless Sensor Networks. In *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 1905–1910, Apr. 2019.

[118] F. Wang, D. Wang, and J. Liu. Traffic-Aware Relay Node Deployment for Data Collection in Wireless Sensor Networks. In *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 1–9, Jun. 2009.

[119] Jin-Woo Kim, José Ramón Ramos Barrado, and Dong-Keun Jeon. Time and Energy Efficient Relay Transmission for Multi-Hop Wireless Sensor Networks. *Sensors*, 16(7):985, 2016.

[120] Hesham Abusaimeh and Mohammad Shkoukani. Modified TCP Protocol for Wireless Sensor Networks. *International Journal of Computer Science Issues (IJCSI)*, 10(2):279–285, Mar. 2013.

[121] H. P. Gupta and S. V. Rao. Demand-Based Coverage and Connectivity-Preserving Routing in Wireless Sensor Networks. *IEEE Systems Journal*, 10(4):1380–1389, Dec. 2016.

[122] Hongya Wang, Jie Jin, Zhijun Wang, and LihChyun Shu. On a Novel Property of the Earliest Deadline First Algorithm. In *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 197–201. IEEE, Jul. 2011.

[123] Vidura Gamini Abhaya, Zahir Tari, Panlop Zeephongsekul, and Albert Y. Zomaya. Performance Analysis of EDF Scheduling in a Multi-Priority Preemptive M/G/1 Queue. *IEEE Transactions on Parallel and Distributed Systems*, 25(8):2149–2158, Aug. 2014.

[124] Christina K. Houben and Wolfgang A. Halang. An Energy-aware Dynamic Scheduling Algorithm for Hard Real-time Systems. In *2014 3rd Mediterranean Conference on Embedded Computing (MECO)*, pages 14–17, Jun. 2014.

[125] R. Gomathi and N. Mahendran. An efficient data packet scheduling schemes in wireless sensor networks. In *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, pages 542–547, Feb. 2015.

[126] Zhibo Wang, Yongpan Liu, Yinan Sun, Yang Li, Daming Zhang, and Huazhong Yang. An Energy-Efficient Heterogeneous Dual-Core Processor for Internet of Things. In *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2301–2304. IEEE, May 2015.

[127] A. Jayanthi and A. Bagavathi. A Low Power Multi-Bit Flip-Flop Design for WSN Nodes. In *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, pages 1418–1421, Feb. 2015.

[128] A. Salleh, K. Mamat, and M. Y. Darus. Integration of wireless sensor network and Web of Things: Security perspective. In *2017 IEEE 8th Control and System Graduate Research Colloquium (ICSGRC)*, pages 138–143, Aug. 2017.

[129] Ali Kashif Bashir, Se-Jung Lim, Sajjad Hussain Chauhdary, and Myong-Soon Park. Energy Efficient In-network RFID Data Filtering Scheme in Wireless Sensor Networks. *Sensors*, 11:7004–7021, Jul. 2011.

[130] A. Yektaparast, F. Nabavi, and A. Sarmast. An improvement on LEACH protocol (Cell-LEACH). In *2012 14th International Conference on Advanced Communication Technology (ICACT)*, pages 992–996, Apr. 2012.

[131] N. A. M. Alduais, J. Abdullah, A. Jamil, and L. Audah. An efficient data collection and dissemination for IoT based WSN. In *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, Oct. 2016.

[132] S. N. Shylaja and M. B. Veena. Real-time monitoring of soil nutrient analysis using WSN. In *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pages 3059–3062, Aug. 2017.

[133] Ravi Kishore Kodali and Aditya Valdas. MQTT Implementation of IoT based Fire Alarm Network. *2018 International Conference on Communication, Computing and Internet of Things (IC3IoT)*, pages 143–146, Mar. 2018.

[134] B. S. Sarierao and A. Prakasarao. Smart Healthcare Monitoring System Using MQTT Protocol. In *2018 3rd International Conference for Convergence in Technology (I2CT)*, pages 1–5, Apr. 2018.

[135] Atmoko Rachmad, Andri and Yang Daoguo. Online Monitoring & Controlling Industrial Arm Robot Using MQTT Protocol. In *2018 IEEE International Conference on Robotics, Biomimetics, and Intelligent Computational Systems (Robionetics)*, pages 12–16, Mar. 2018.

[136] A. Jangra and N. Bhardwaj. RAINs: Recursive algorithm for impostor node selection to balance the load in MANETs. In *2014 International Conference on Communication and Signal Processing*, pages 1369–1374, Apr. 2014.

[137] Arora Vishal, Kumar, Sharma Vishal, and Sachdeva Monika. On QoS evaluation for ZigBee incorporated Wireless Sensor Network (IEEE 802.15.4) using mobile sensor nodes. *Journal of King Saud University -Computer and Information Sciences*, Oct. 2018.

[138] H. Zhou, X. Guan, and C. Wu. Reliable Transport with Memory Consideration in Wireless Sensor Networks. In *2008 IEEE International Conference on Communications*, pages 2819–2824, May 2008.

[139] Muhammad Farooq-i-Azam and Muhammad Ayyaz Naeem. Location and Position Estimation in Wireless Sensor Networks. *arXiv e-prints*, abs/1611.03420, Oct. 2016.

[140] Y Kan, S Chiang, C Lin, and A Hardware. A GPS Anchor Node for Outdoor Wireless Sensor Network Applications. In *IEEE International Symposium on Radio-Frequency Integration Technology (RFIT)*, pages 40–43, Jan. 2009.

[141] Yonggang Xu, Haohao Qi, Tianyi Xu, Qianqian Hua, Hongsheng Yin, and Gang Hua. Queue models for wireless sensor networks based on random early detection . *Peer-to-Peer Networking and Applications*, 12:1539–1549, May 2019.

[142] Saima Abdullah and Kun Yang. An energy-efficient message scheduling algorithm in Internet of Things environment. In *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 311–316. IEEE, Jul. 2013.

[143] Reza Rasouli, Mahmood Ahmadi, and Ali Ahmadvand. Energy Consumption Estimation in Clustered Wireless Sensor Networks Using M/M/1 Queuing Model. *International Journal of Wireless and Mobile Networks*, 5:15–31, Feb. 2013.

[144] Lulu Liang, Deyun Gao, and Victor C.M. Leung. Queue-based congestion detection and multistage rate control in event-driven wireless sensor networks. *Wireless Communtions and Mobile Computing*, 14:818–830, Jun. 2014.

[145] Muhammad Omer Farooq and Thomas Kunz. Operating systems for wireless sensor networks: A survey. *Sensors*, 11(6):5900–5930, 2011.

[146] I. Keslassy G. Appenzeller and N. McKeown. Sizing Router Buffers. *ACM SIGCOMM Computer Communication Review*, 34(4):281–292, Aug. 2004.

[147] Darijo Raca, Ahmed H Zahran, and Cormac J Sreenan. Sizing Network Buffers : A HTTP Adaptive Streaming Perspective. *International Conference on Future Internet of Things and Cloud Workshops*, Aug. 2016.

[148] Yousi Zheng, Lei Wan, Zhi Sun, and Shunliang Mei. A Long Range DV-Hop Localization Algorithm with Placement Strategy in Wireless Sensor Networks. In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–5. IEEE, Oct. 2008.

[149] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pages 3005–3014, 2000.

[150] XBee Modules. XBee ® /XBee-PRO ® RF Modules. In *Product Manual v1. xEx-802.15.4 Protocol*. Sparkfun (online), 2009.

[151] Azer Bestavros Smaragdakis, Georgios, Ibrahim Matta. SEP : A Stable Election Protocol for Clustered Heterogeneous Wireless Sensor Networks. In *In proceedings of the 2nd international workshop on SANPA*, pages 1–11, Jul. 2004.

[152] R. Kasana, S. Kumar, O. Kaiwartya, W. Yan, Y. Cao, and A. H. Abdullah. Location error resilient geographical routing for vehicular ad-hoc networks. *IET Intelligent Transport Systems*, 11(8):450–458, Oct. 2017.

[153] Z. Abedin, S. Paul, S. Akhter, K. N. Siddiquee, M. S. Hossain, and K. Andersson. Selection of Energy Efficient Routing Protocol for Irrigation Enabled by Wireless Sensor Network. In *2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops)*, pages 75–81, Nov. 2017.

[154] I. Hou. Packet Scheduling for Real-Time Surveillance in Multihop Wireless Sensor Networks With Lossy Channels. *IEEE Transactions on Wireless Communications*, 14(2):1071–1079, Feb. 2015.

[155] T. F. Wykret, L. H. A. Correia, D. F. Macedo, J. C. Giacomin, and L. T. Andrade. Evaluation and avoidance of interference in WSN: A multi-radio node prototype using Dynamic Spectrum Allocation. In *2013 IFIP Wireless Days (WD)*, pages 1–3, Nov. 2013.

[156] Campbell Carlene CE, Loo Kok-Keong, Gemikonakli Orhan, Khan Shafiullah, and Singh Dhananjay. Multi-Channel Distributed Coordinated Function over Single Radio in Wireless Sensor Networks. *Sensors*, 11(1):964–991, Jan. 2011.

[157] J. Ma, D. Yang, H. Wang, and M. Gidlund. An Efficient Retransmission Scheme for Reliable End-to-End Wireless Communication Over WSANs. *IEEE Access*, 6:49838–49849, Aug. 2018.

[158] Baumann Rainer, Heimlicher Simon, Strasser Mario, and Andreas Weibel. A Survey on Routing Metrics. *TIK Rep. 262 ETH-Zent*, pages 1–53, Feb. 2007.

[159] Shiliang Xiao, Lebing Pan, Jianpo Liu, Baoqing Li, and Xiaobing Yuan. Distributed Broadcast with Minimum Latency in Asynchronous Wireless Sensor Networks under SINR-Based Interference. *International Journal of Distributed Sensor Networks*, 9(11):506797, Nov. 2013.

[160] Timo Kumberg, Christian Schindelhauer, and Leonhard Reindl. Exploiting Concurrent Wake-Up Transmissions Using Beat Frequencies. *Sensors (Basel)*, 17:1717, Jul. 2017.

[161] Jims Marchang, Bogdan Ghita, and David Lancaster. Location based transmission using a neighbour aware with optimized EIFS MAC for ad hoc networks. *Ad Hoc Networks*, 63:62–78, Aug. 2017.

[162] Luigi Iannone and Serge Fdida. MRS: A Simple Cross-Layer Heuristic to Improve Throughput Capacity in Wireless Mesh Networks. In *CoNEXT 2005 - ACM conference on emerging network experiment and technology*, pages 21–30, Toulouse, France, Oct. 2005. ACM.

[163] Shahadat Hossain, Rohan Monteiro, Allaa R. Hilal, and Otman Basir. Fault Tolerant Wireless Sensor Networks using Adaptive Partitioning. *Procedia Computer Science*, 10:927 – 932, Jun. 2012.

[164] T. Matsui and H. Nishi. ECORS: Energy consumption-oriented route selection for wireless sensor network. In *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, Jul. 2016.

[165] Ma Xiliang and Mao Ruiqing. Method for Effectively Utilizing Node Energy of WSN for Coal Mine Robot. *Journal of Robotics*, 2018:8, Jan. 2018.

[166] A. E. Fawzy, A. Amer, M. Shokair, and W. Saad. Proposed intermittent cluster head selection scheme for efficient energy consumption in WSNs. In *2017 34th National Radio Science Conference (NRSC)*, pages 275–283, Mar. 2017.

[167] A. Kanawaday and A. Sane. Machine learning for predictive maintenance of industrial machines using IoT sensor data. In *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pages 87–90, Nov. 2017.

[168] A. D. Deshmukh and U. B. Shinde. A low cost environment monitoring system using raspberry Pi and arduino with Zigbee. In *2016 International Conference on Inventive Computation Technologies (ICICT)*, volume 3, pages 1–6, Aug. 2016.

[169] Nagy Tamas and Gingl Zoltan. Low-cost photoplethysmograph solutions using the Raspberry Pi. In *2013 IEEE 14th International Symposium on Computational Intelligence and Informatics (CINTI)*, Nov. 2013.

[170] Firas Abdulsattar, Dennis Cooper, Hector Iacovides, and Shanying Zhang. Turbulent Flow Development inside a Rotating Two-Pass Square Duct with Porous Blocks. In *23rd ISABE Conference, Manchester, United Kingdom*, Sept. 2017.

[171] N. Aroon. Study of using MQTT cloud platform for remotely control robot and GPS tracking. In *2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Jun. 2016.

[172] S. Lee, H. Kim, D. Hong, and H. Ju. Correlation analysis of MQTT loss and delay according to QoS level. In *The International Conference on Information Networking 2013 (ICOIN)*, Jan. 2013.

[173] A. Bhawiyuga, D. P. Kartikasari, and E. S. Pramukantoro. A publish subscribe based middleware for enabling real time web access on constrained device. In *2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Oct. 2017.

[174] Quality of Service 0,1 & 2 - MQTT Essentials. https://www.hivemq.com/blog/mqtt-essentials-part-6-mqtt-quality-of-service-levels/. Accessed: 03-06-2019.

[175] R. K. Kodali and V. S. K. Gorantla. Weather tracking system using MQTT and SQLite. In *2017 3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, Dec. 2017.

[176] Jun-Hong Park, Hyeong-Su Kim, and Won-Tae Kim. DM-MQTT: An Efficient MQTT Based on SDN Multicast for Massive IoT Communications. *Sensors*, 18(9), Sept. 2018.

[177] Istabraq M. Al-Joboury and Emad H. Al-Hemiary. *F2CDM: Internet of Things for Healthcare Network Based Fog-to-Cloud and Data-in-Motion Using MQTT Protocol*, volume 10542. Springer International Publishing, Cham, 2017.

[178] Gu Qinlong, Cheng Xingmei, Tao Weiwei, and Yao Minghai. Study and application of SQLite embedded database system based on Windows cE. In *The 2nd International Conference on Information Science and Engineering*, pages 6920–6923, Dec. 2010.

[179] L. Junyan, X. Shiguo, and L. Yijie. Application Research of Embedded Database SQLite. In *2009 International Forum on Information Technology and Applications*, May 2009.

[180] Jihong Zhang and Xiaoquan Chen. Research and Design of Embedded Wireless Meal Ordering System Based on SQLite. *Physics Procedia*, 25:583 – 587, Mar. 2012.

[181] Chunyue Bi. Research and Application of SQLite Embedded Database Technology. *WSEAS Transactions on Computers*, 8(1):83–92, Jan. 2009.

[182] A. Sinha, S. Sharma, and M. R. Mahboob. An Internet of Things based prototype for smart appliance control. In *2017 International Conference on Computing, Communication and Automation (ICCCA)*, May 2017.

[183] V. Patchava, H. B. Kandala, and P. R. Babu. A Smart Home Automation technique with Raspberry Pi using IoT. In *2015 International Conference on Smart Sensors and Systems (IC-SSS)*, pages 1–4, Dec. 2015.

[184] Jose L. Ferreira, Marconi Oliveira Jr., Veronica F. Parreira, Nadja C. Pereira, Flavio H. Vasconcelos, and Carlos J. Tierra-Criollo. Maximum respiratory pressure measuring system: calibration and evaluation of uncertainty. *Sba: Controle & Automacao*, 21:588 – 597, Dec. 2010.

[185] Custard Pi 3A. http://www.sf-innovations.co.uk/custard-pi-3.html. Accessed:10-05-2019.

[186] Relative, absolute and differential pressure transmitter 663. http://www.sts-italia.it/2011/pdf/663.pdf. Accessed: 29-08-2019.

[187] H. S. Raju and S. Shenoy. Real-time remote monitoring and operation of industrial devices using IoT and cloud. In *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pages 324–329, Dec. 2016.